A Local Method for Identifying Causal Relations under Markov Equivalence

Zhuangyan Fang\textsuperscript{1} Yue Liu\textsuperscript{2} Zhi Geng\textsuperscript{1} Yangbo He\textsuperscript{1}\textsuperscript{*}
\textsuperscript{1}Peking University \textsuperscript{2}Huawei Noah’s Ark Lab

February 26, 2021

Abstract

Causality is important for designing interpretable and robust methods in artificial intelligence research. We propose a local approach to identify whether a variable is a cause of a given target based on causal graphical models of directed acyclic graphs (DAGs). In general, the causal relation between two variables may not be identifiable from observational data as many causal DAGs encoding different causal relations are Markov equivalent. In this paper, we first introduce a sufficient and necessary graphical condition to check the existence of a causal path from a variable to a target in every Markov equivalent DAG. Next, we provide local criteria for identifying whether the variable is a cause/non-cause of the target. Finally, we propose a local learning algorithm for this causal query via learning local structure of the variable and some additional statistical independence tests related to the target. Simulation studies show that our local algorithm is efficient and effective, compared with other state-of-art methods.

1 Introduction

In many observational studies, the main purposes are to study whether a treatment variable is a cause of a target variable, or to further identify the causes/non-causes of a specified target variable or the effects/non-effects of a given treatment. For examples, in a clinical study we may concern about which symptoms are the side effects caused by using a new drug and which are not. Causality is also important for designing interpretable and robust methods in artificial intelligence research (Miller 2019), and has been

\textsuperscript{*}Correspondence to: heyb@pku.edu.cn.
used in many fields of artificial intelligence, such as causal transfer learning (Zhang et al., 2020; Bengio et al., 2020), and causality-based algorithmic fairness (Kusner et al., 2017; Wu et al., 2019). Directed acyclic graphs (DAGs) can be used to represent causal relationships among variables (Pearl, 2009).

If a variable $X$ has a directed path to another variable $Y$, then $X$ is a cause of $Y$ and $Y$ is an effect of $X$. From observational data, however, instead of an exact causal DAG, we generally learn a Markov equivalence class of DAGs which can be represented by a completely partially directed acyclic graph (CPDAG). The undirected edges in a CPDAG imply that some causal relations among variables can not be read from the graph directly.

Consider a Markov equivalence class of DAGs learned from observational data. A variable $X$ is a definite cause of a target $Y$ if $X$ is always a cause of $Y$ in every equivalent DAG, and a variable $X$ is a definite non-cause of $Y$ if $X$ is never a cause of $Y$ in any DAG in the class. If $X$ is neither a definite cause nor a definite non-cause of $Y$, $X$ is called a possible cause of $Y$.

Some approaches can be used to identify the causal relation between a treatment and a target. An intuitive approach is first to learn a Markov equivalence class from observational data, and then enumerate all DAGs in the class to check whether the treatment is definitely or definitely not the cause of the target in all of these equivalent DAGs. However, the intuitive approach is inefficient when the number of DAGs in the learned Markov equivalence class is large (He et al., 2015).

Another way is to check the paths from the treatment to the target in a CPDAG. It has been shown that the treatment is a definite non-cause of the target if and only if there is no partially directed path from the treatment to the target (see, e.g. Zhang, 2006; Perković et al., 2017). Given a CPDAG, Roumpelaki et al. (2016) also introduces a sufficient condition for identifying definite causes. However, the necessity of this condition remains a conjecture (Zhang, 2006; Mooij and Claassen, 2020) and the corresponding approach is usually inefficient since it needs to learn an entire CPDAG first.

The third approach is to estimate the causal effect of the treatment on the given target (Maathuis et al., 2009; Perković et al., 2017; Nandy et al., 2017; Fang and He, 2020; Liu et al., 2020; Witte et al., 2020; Guo and Perković, 2020). This approach, which is called causal-effect-based method, determines whether a variable is a cause of another by testing whether all possible causal effects are zeros/non-zeros. However, this method requires

\[1\] We note that, the recent progresses in identifying the causal relation between two variables indeed provide an opportunity to learn an exact DAG. However, such methods need to pose additional distributional conditions (Shimizu et al., 2006; Zhang and Hyvärinen, 2009; Shimizu et al., 2011; Peters and Bühlmann, 2013; Peters et al., 2014).
additional model assumptions to estimate causal effects, and it would be difficult to choose a suitable significance level in practice if the number of possible effects is large.

In this paper, we study the problem of locally identifying causal relations under Markov equivalence with assumption that there is no hidden variable or selection bias. That is, given a pair of treatment and target variables, we intend to decide whether the treatment is a definite cause, a possible cause or a definite non-cause of the target only based on a local induced subgraph and a few independence tests related to the treatment without learning an entire CPDAG. This local approach is usually much efficient than the global ones that need an entire CPDAG, especially when the underlying causal graph is large.

To this end, we first discuss the existence of a causal path from one variable to another given a CPDAG, and prove the necessity of the condition in Roumpelaki et al. (2016) for CPDAGs. This yields a sufficient and necessary graphical condition to check the existence of a causal path. Next, we propose local identification criteria for definite causes, possible causes or definite non-causes. These criteria depend only on the induced subgraph of the true CPDAG over the adjacent variables of the treatment as well as some queries about d-separations, thus directly lead to a local learning algorithm for learning types of causal relations. For the completeness of the paper, a global algorithm for learning types of causal relations is also provided. Finally, we compare experimentally the proposed local learning method with the global and the causal-effect-based methods, and show the efficiency and efficacy of our local method.

The rest of the paper is organized as follows. In Section 2, we introduce the notation, the definitions and the assumptions used in our work. We provide a sufficient and necessary graphical condition to check the existence of a causal path from a treatment to a target based on a CPDAG in Section 3. In Section 4, we formally present our local characterizations for definite causal relations and definite non-causal relations. The local learning algorithm for identifying the type of causal relation is given in Section 5. Experimental results are illustrated in Section 6. In Section 7, we discuss the applications and some possible extensions of our work. Finally, Some graph terminology, global and causal-effect-based algorithms for learning causal relations, and the detailed proofs, are shown in Appendix A, B and C, respectively.
2 Background

In this section, we introduce basic concepts of causal graphical models, and the assumptions for causal learning. We use \( \text{pa}(S, G) \), \( \text{ch}(S, G) \), \( \text{sib}(S, G) \), \( \text{adj}(S, G) \), \( \text{an}(S, G) \) and \( \text{de}(S, G) \) to denote the union of parents, children, siblings (or undirected neighbors), adjacent vertices, ancestors, and descendants of each variable in set \( S \) in \( G \), respectively, where \( G = (V, E) \) can be a directed, an undirected, or a partially directed graph. The basic graph terminology can be found in A. As a convention, we regard a vertex as an ancestor and a descendant of itself. If \( S = \{X\} \) is a singleton set, we will replace \( S \) by \( X \) for ease of presentation. Let \( G \) be a causal acyclic directed graph (causal DAG) and \( X \) be a vertex in \( G \), the vertices in \( \text{an}(X, G) \setminus X \) are causes of \( X \), and the vertices in \( \text{pa}(X, G) \) are direct causes of \( X \). If \( X \) is a cause of \( Y \), then the directed paths from \( X \) to \( Y \) are called causal paths.

2.1 Causal DAG Models

The notion of \( d \)-separation induces a set of conditional independence relations encoded in a DAG (Pearl, 1988). Let \( G \) be a DAG and \( \pi = (X = X_0, X_1, ..., X_n = Y) \) be a path from \( X \) to \( Y \) in \( G \). An intermediate vertex \( X_i \) is a collider on \( \pi \) if \( X_{i-1} \rightarrow X_i \) and \( X_i \leftarrow X_{i+1} \), otherwise, \( X_i \) is a non-collider on \( \pi \). For three distinct vertices \( X_i, X_j \) and \( X_k \), if \( X_i \rightarrow X_j \leftarrow X_k \) and \( X_i \) is not adjacent to \( X_k \) in \( G \), then the triple \( (X_i, X_j, X_k) \) is called a \( v \)-structure collided on \( X_j \) in \( G \). Given \( Z \subseteq V \), we say \( \pi \) is \( d \)-connected (or active) given \( Z \) if \( Z \) does not contain any endpoint or non-collider on the path and every collider on the path has a descendant in \( Z \). If \( \pi \) is not \( d \)-connected given \( Z \), then \( \pi \) is blocked by \( Z \). For pairwise disjoint sets \( X, Y, Z \subseteq V, X \) and \( Y \) are \( d \)-separated by \( Z \) (denoted by \( X \perp \perp Y \mid Z \) if and only if every path between some \( X \in X \) and \( Y \in Y \) is blocked by \( Z \).

Let \( J_G \) be the set of \( d \)-separation relations read off from a DAG \( G \). Two DAGs \( G_1 \) and \( G_2 \) are Markov equivalent if \( J_{G_1} = J_{G_2} \). Pearl et al. (1989) have shown that two DAGs are equivalent if and only if they have the same skeleton and the same \( v \)-structures. A Markov equivalence class or simply equivalence class, denoted by \([G]\), contains all DAGs equivalent to \( G \). A Markov equivalence class \([G]\) can be uniquely represented by a partially directed graph called completely partially directed acyclic graph (CPDAG) \( G^* \). Two vertices are adjacent in \( G^* \) if and only if they are adjacent in \( G \) and a directed edge occurs in \( G^* \) if and only if it appears in every DAG in \([G]\). For the ease of presentation, we will also use \([G^*]\) to represent the Markov equivalence class represented by \( G^* \) below. Given a
CPDAG $\mathcal{G}^*$, we use the $\mathcal{G}^*_u$ and $\mathcal{G}^*_d$ to denote the undirected subgraph and the directed subgraph of $\mathcal{G}^*$, respectively. The former is defined as the undirected graph resulted by removing all directed edges in $\mathcal{G}^*$ and the later is the directed graph obtained by removing undirected edges. [Andersson et al. (1997)] proved that (1) the undirected subgraph $\mathcal{G}^*_u$ of $\mathcal{G}^*$ is the union of disjoint connected chordal graphs (the definition of chordal graph is provided in A), and (2) every partially directed cycle in $\mathcal{G}^*$ is an undirected cycle, that is, none of the partially directed cycles in $\mathcal{G}^*$ contains a directed edge. Each isolated connected undirected subgraph of $\mathcal{G}^*_u$ is called a chain component of $\mathcal{G}^*$ [Andersson et al. (1997), Lauritzen and Richardson (2002)].

For a given distribution $P$, we use $X \perp \perp P Y \mid Z$ to denote that $X$ is independent of $Y$ given $Z$ with respect to $P$. Let $\mathcal{J}_P$ be the set of all such (conditional) independencies in $P$. In this paper, our main results are based on the following assumptions: *causal Markov assumption*, which states that $X \perp Y \mid Z$ in $\mathcal{J}_G$ implies $X \perp Y \mid Z$ in $\mathcal{J}_P$; *causal faithfulness assumption*, which states that $X \perp P Y \mid Z$ in $\mathcal{J}_P$ implies $X \perp Y \mid Z$ in $\mathcal{J}_G$; and the assumption that there is no hidden variable or selection bias. A distribution $P$ is called Markov and faithful to a DAG $\mathcal{G}$ if $P$ and $\mathcal{G}$ satisfy the causal Markov assumption and the causal faithfulness assumption. A causal DAG model consists of a DAG $\mathcal{G}$ and a joint distribution $P$ over a common vertex set $\mathcal{V}$ such that $P$ satisfies the causal Markov assumption with respect to $\mathcal{G}$. $\mathcal{G}$ is called the causal structure of the model and $P$ is called the observational distribution (or simply distribution) [Hauser and Bühlmann (2012)].

2.2 Global and Local Causal Structure Learning

Causal structure learning methods try to recover the causal structure from data. The global causal structure learning focuses on learning an entire causal structure over all variables while the local causal structure learning discusses how to recover a part of the underlying causal structure.

Existing approaches for learning global causal structures roughly fall into two classes: constraint-based and score-based methods. Constraint-based methods, such as the PC algorithm [Spirtes and Glymour (1991)] and the stable PC algorithm [Colombo and Maathuis (2014)], use conditional independence tests to find causal skeleton and then determine the edge directions according to a series of orientation rules [Meek (1995)]. Under the causal Markov and causal faithfulness assumptions, the constraint-based methods can identify causal graphs up to a Markov equivalence class. On the other hand, score-based methods, such as exact search algorithms like dynamic programming [Koivisto and Sood (2004); Singh and Moore (2005)] and A* [Yuan (2001)]
et al., 2011; Xiang and Kim, 2013), heuristic search algorithms like GES (Chickering, 2002b), and gradient-based methods like NOTEARS (Zheng et al., 2018), evaluate the candidate graphs with a predefined score function and search for DAGs or CPDAGs with the optimal score.

Local learning algorithms usually learn the Markov blanket (see, e.g., Tsamardinos et al., 2003; Tsamardinos and Aliferis, 2003; Fu and Desmarais, 2010) or the parent and child set of a given target (see, e.g., Wang et al., 2014; Gao and Ji, 2015; Liu et al., 2019). Recently, Liu et al. (2020b, Algorithm 3) extended the MB-by-MB algorithm (Wang et al., 2014) to learn the chain component containing a given target and the directed edges surrounding the chain component. This variant of MB-by-MB can thus learn the subgraph of the CPDAG over the target and its neighbors, that is, the parents, siblings and children of the target in the CPDAG.

3 An Anatomy of Causal Relations

In this section, we provide a sufficient and necessary condition to identify definite causal relations, and show that the definite causal relations can be divided into two subtypes: explicit and implicit causal relations.

3.1 Graphical Criteria for Identifying Types of Causal Relations

As mentioned in Section 1, given a CPDAG, a variable $X$ is a definite non-cause of another variable $Y$ if and only if there is no partially directed path from $X$ to $Y$ (Zhang, 2006; Perković et al., 2017). Roumpelaki et al. (2016, Theorem 3.1) proved that the treatment is a definite cause of the target if there is a directed path from the treatment to the target or the treatment has two chordless partially directed path to the target on which two vertices adjacent to the treatment are distinct and non-adjacent. In the section, we will show this condition is also necessary, and before that, a concept of critical set is introduced as follows.

**Definition 1** (Critical Set). (Fang and He, 2020, Definition 2) Let $G^*$ be a CPDAG. $X$ and $Y$ are two distinct vertices in $G^*$. The critical set of $X$ with respect to $Y$ in $G^*$ consists of all adjacent vertices of $X$ lying on at least one chordless partially directed path from $X$ to $Y$.

The definition of chordless partially directed path can be found in A. With Definition 1, we have the following lemma.
Lemma 1. Let $G^*$ be a CPDAG. For any two distinct vertices $X$ and $Y$ in $G^*$, $X$ is a definite cause of $Y$ in the underlying DAG if and only if the critical set of $X$ with respect to $Y$ in $G^*$ always contains a child of $X$ in every DAG $G \in [G^*]$.

Lemma 1 follows from Lemma 2 in Fang and He (2020). It gives a sufficient and necessary condition to decide whether $X$ is a definite cause of $Y$. However, checking the condition given in Lemma 1 also requires to enumerate all equivalent DAGs. To mitigate this problem, we discuss a graph characteristic of critical set in the corresponding CPDAG.

Lemma 2. Let $G^*$ be a CPDAG and $X, Y$ be two distinct vertices in $G^*$. Denote by $C$ the critical set of $X$ with respect to $Y$ in $G^*$, then $C \cap \text{ch}(X, G) = \emptyset$ for some $G \in [G^*]$ if and only if $C = \emptyset$, or $C$ induces a complete subgraph of $G^*$ but $C \cap \text{ch}(X, G^*) = \emptyset$.

Based on Lemmas 1 and 2 we have the desired sufficient and necessary graphical criterion.

Theorem 1. Suppose that $G^*$ is a CPDAG, $X, Y$ are two distinct vertices in $G^*$, and $C$ is the critical set of $X$ with respect to $Y$ in $G^*$. Then, $X$ is a definite cause of $Y$ if and only if $C \cap \text{ch}(X, G^*) \neq \emptyset$, or $C$ is non-empty and induces an incomplete subgraph of $G^*$.

The sufficiency of the condition in Theorem 1 has been extended to other types of causal graphs by Roumpelaki et al. (2016) and Mooij and Claassen (2020).² With the help of Theorem 1, we can identify the type of causal relation based on a learned CPDAG by enumerating paths and finding critical sets. Below, we give an example to illustrate this idea.

Example 1. Consider the respiratory disease network shown in Figure 1. Let smoking be the treatment and dyspnoea be the target. From Figure 1(b) we can see that the partially directed paths from smoking to dyspnoea are Smok → Lung → Either → Dysp and Smok → Bronc → Dysp. Therefore, the critical set of smoking with respect to dyspnoea is $\{\text{Lung, Bronc}\}$. As Lung and Bronc are not adjacent, by Theorem 1 smoking is a definite cause of dyspnoea. Similarly, the critical set of lung cancer with respect to dyspnoea is $\{\text{Smok, Either}\}$. Since Either is a child of Lung, lung cancer is also a definite cause of dyspnoea.

²We note that, although Roumpelaki et al. (2016, Theorem 3.1) also claimed that they have proved the necessity, their proof is flawed. As mentioned by Mooij and Claassen (2020), the last part of the proof appears to be incomplete. How to prove the necessity for more general types of causal graphs remains an open problem (Zhang, 2006).
Figure 1: This example is adapted from the ASIA network. The original network structure and related parameters can be found in Lauritzen and Spiegelhalter (1988). Figure 1(a) shows the true underlying causal DAG, and Figure 1(b) shows the corresponding CPDAG. Figure 1(c) enumerates all equivalent DAGs in the Markov equivalence class.

3.2 Explicit and Implicit Causal Relations

We now study the properties of definite causal relations, and show that definite causal relations can be divided into two subtypes based on the existence of causal paths in a CPDAG. The results in this section are of key importance to build local characterizations in Section 4, and are also useful for developing an efficient global learning algorithm.

**Proposition 1.** For two distinct vertices $X$ and $Y$, if $X$ is a definite cause of $Y$, then $X$ and $Y$ are not in the same chain component.

Given a target variable $Y$, Proposition 1 shows that $Y$ and its definite causes do not appear in the same chain component. Thus, if a treatment $X$ is a definite cause of a target $Y$, then in $G^*$ there must be a partially directed path from $X$ to $Y$ which contains a directed edge. On the other hand, for two distinct vertices lying in the same chain component, we have,

**Proposition 2.** Two distinct vertices $X$ and $Y$ are possible causes of each other if and only if they are in the same chain component.

Recall that in Figure 1(b) both Smok and Lung are definite causes of Dysp. However, in the CPDAG there exists a directed path from Lung to Dysp while no directed path exists from Smok to Dysp. That is, the cause Lung of Dysp is explicit and the cause Smok of Dysp is implicit in the CPDAG. This difference motivates us to introduce the following two concepts.
Definition 2 (Explicit Cause). A variable $X$ is an explicit cause of $Y$ if $X$ is a definite cause of $Y$ and there is a common causal path from $X$ to $Y$ in every DAG in the Markov equivalence class represented by a CPDAG $G^*$. 

Since there is a common causal path from an explicit cause $X$ to the target $Y$ in every DAG in the Markov equivalence class represented by $G^*$, there is a directed path from $X$ to $Y$ in $G^*$. As a convention, we regard $X$ as an explicit cause of itself.

Definition 3 (Implicit Cause). A variable $X$ is an implicit cause of $Y$ if $X$ is a definite cause of $Y$ and there is no common causal path from $X$ to $Y$ in every DAG in the Markov equivalence class represented by a CPDAG $G^*$. 

We notice that $X$ is a definite cause of $Y$ if only if it satisfies one of two conditions given in Theorem 1. The first condition, $C \cap ch(X, G^*) \neq \emptyset$, is the sufficient and necessary condition for identifying explicit causes, while the second condition corresponds to implicit causes. In Section 4, we will exploit this difference between explicit and implicit causes to develop local characterizations for both of them. Below, we give an example to illustrate them.

Example 2. Consider the causes of the target variable $Y$ based on the CPDAG $G^*$ in Figure 2. It is clear that all the variables other than $Y$ are definite or possible causes of $Y$. Obviously, $\{E, D, F\}$ are explicit causes of $Y$. For $B$, since $B \rightarrow E \rightarrow Y$, $B \rightarrow D \rightarrow Y$ and $B \rightarrow G \rightarrow F \rightarrow Y$ are chordless partially directed paths, the critical set of $B$ with respect to $Y$ is $\{E, D, G\}$. As the induced subgraph of $G^*$ over $\{E, D, G\}$ is not complete, $B$ is a definite cause of $Y$, and $B$ is also implicit. Similarly, $G$ is another implicit cause of $Y$. For $X$ and $A$, the critical set of $X$ and $A$ with respect to $Y$ are $\{B, D, G\}$ and $\{X, G\}$, respectively. Since the corresponding induced

Figure 2: An example for identifying the types of causal relations
subgraphs are complete, by Theorem 1, $X$ and $A$ are not implicit causes of $Y$. Thus, they are possible causes of $Y$.

Despite the difference, explicit and implicit causes also have some interesting connections. The following Proposition 3 proves that the existence of an implicit cause implies the existence of at least two explicit causes.

**Proposition 3.** Let $G^*$ be a CPDAG. $X$ and $Y$ are two distinct vertices that belong to different chain components. If $X$ is the only explicit cause of $Y$ in the chain component to which $X$ belongs, then every vertex in this chain component, except $X$, is a possible cause of $Y$.

4 Local Characterizations of the Types of Causal Relations

In this section, we introduce the theoretical results on locally characterizing different types of causal relations. Our local characterizations depend on the induced subgraph of the true CPDAG over the treatment’s neighbors as well as some queries about d-separation relations. The first result is about definite non-causal relations, as given in Theorem 2.

**Theorem 2.** Let $G^*$ be a CPDAG. For any two distinct vertices $X$ and $Y$ in $G^*$, $X$ is a definite non-cause of $Y$ if and only if $X \perp \!\!\!\perp Y \mid pa(X, G^*)$ holds.

Theorem 2 introduces a local characterization for definite non-causal relations, which is based on the local structure around the treatment $X$ and a single d-separation claim. The d-separation claim $X \perp \!\!\!\perp Y \mid pa(X, G^*)$ is similar to the following well-known result called local Markov property of a causal DAG model: any variable is d-separated from its non-descendants given its parents. The difference is that in our local characterization, only parents of $X$ in the CPDAG are included in the separation set, and we rule out the siblings of $X$ even if they may be the parents of $X$ in the true causal DAG. Since in a causal DAG, the non-descendants of a variable are those which are definitely not caused by the variable, Theorem 2 can be regarded as an extension of the local Markov property to CPDAGs.

Following Theorem 2, we can distinguish the definite causes and the possible causes from the definite non-causes with a local causal structure query and a d-separation query. Next, we characterize explicit and implicit causal relations locally in Theorem 3 and Theorem 4, respectively, which together characterize definite causal relations.
Theorem 3. Let $G^*$ be a CPDAG. For any two distinct vertices $X$ and $Y$ in $G^*$, $X$ is an explicit cause of $Y$ if and only if $X \not\perp Y \mid pa(X, G^*) \cup \text{sib}(X, G^*)$ holds.

The local characterization in Theorem 3 includes a single d-separation claim, $X \not\perp Y \mid pa(X, G^*) \cup \text{sib}(X, G^*)$, which means the set $pa(X, G^*) \cup \text{sib}(X, G^*)$ cannot block all paths from $X$ to $Y$. In the proof of this theorem, we show that this claim is equivalent to that there exists at least one path from $X$ to $Y$ in $G^*$ on which the node adjacent to $X$ is a child of $X$. Based on Maathuis and Colombo (2015, Lemma 7.2) and Perković et al. (2017, Lemma B.1), this implies that there is a directed path from $X$ to $Y$ in $G^*$.

Theorem 4. Suppose that $G^*$ is a CPDAG and $M$ is the set of maximal cliques of the induced subgraph of $G^*$ over $\text{sib}(X, G^*)$. Then, $X$ is an implicit cause of $Y$ if and only if $X \not\perp \perp Y \mid pa(X, G^*) \cup \text{sib}(X, G^*)$ and $X \not\perp Y \mid pa(X, G^*) \cup M$ for any $M \in M$.

The definition of maximal clique is given in [A]. In Theorem 4, the first condition $X \not\perp Y \mid pa(X, G^*) \cup \text{sib}(X, G^*)$ makes sure that $X$ is not an explicit cause of $Y$ and the second condition, which is $X \not\perp Y \mid pa(X, G^*) \cup M$ for any $M \in M$, guarantees that $X$ is not a possible cause of $Y$. These two conditions in Theorem 4 are local in the sense that both $\text{sib}(X, G^*)$ and $pa(X, G^*)$ are subsets of $X$’s neighbors in $G^*$, and a maximal clique $M$ is also a subset of $\text{sib}(X, G^*)$. Once we obtain the induced subgraph of $G^*$ over $\text{adj}(X, G^*)$, we can know $\text{sib}(X, G^*)$ and $M$, and thus the conditional independence queries can be answered accordingly if we have the oracles.

As mentioned in Section 3.2, definite causes consists of both explicit causes and implicit causes. Therefore, Theorems 3 and 4 give a sound and complete local characterization of definite causal relations as follows.

Corollary 1. Suppose that $G^*$ is a CPDAG and $M$ is the set of maximal cliques of the induced subgraph of $G^*$ over $\text{sib}(X, G^*)$. Then, $X$ is a definite cause of $Y$ if and only if $X \not\perp Y \mid pa(X, G^*) \cup \text{sib}(X, G^*)$ or $X \not\perp Y \mid pa(X, G^*) \cup M$ for any $M \in M$.

Together with Theorem 2, Corollary 4 can be used to identify the definite causal relations and the definite non-causal relations. This result is local in the sense that it only depends on the local structure around the treatment $X$ and a limited number of d-separation queries. When data is available in practice, d-separation queries can be answered by performing statistical independence tests. Thus, local characterizations are particularly meaningful for identifying the types of causal relations from observational data.
5 A Local Learning Algorithm

In this section, we discuss how to learn the type of causal relation from observational data. A local algorithm, which exploits the local characterizations in Section 4 directly, is provided in this section.

The main procedure of our local algorithm is summarized in Algorithm 1. The input of Algorithm 1 consists of \( \text{pa}(X, G^*) \), the induced subgraph of \( G^* \) over \( \text{sib}(X, G^*) \), and some independence oracles. The first two arguments, \( \text{pa}(X, G^*) \) and the induced subgraph over \( \text{sib}(X, G^*) \), can be learned locally by using the variant of the MB-by-MB algorithm proposed by Liu et al. (2020b, Algorithm 3), which is designed for learning the chain component containing a given target variable and the directed edges connected to the variables in the chain component. The third argument (the independence oracles), as discussed in Section 4, can be replaced by statistical independence tests in practice. Overall, the procedure given in Algorithm 1 is a direct application of the local characterizations in Theorems 2, 3 and 4 and thus we have,

**Theorem 5.** Assuming that the true causal DAG is in the Markov equivalence class represented by \( G^* \) and the independence oracles are faithful to the true DAG, then the local ITC (Algorithm 1) can correctly identify the type of causal relation between \( X \) and \( Y \).

---

**Algorithm 1** A local algorithm for identifying the type of causal relation (local ITC)

**Require:** A treatment \( X \), a target \( Y \), \( \text{pa}(X, G^*) \), the induced subgraph of \( G^* \) over \( \text{sib}(X, G^*) \), and independence oracles.

**Ensure:** The type of causal relation between \( X \) and \( Y \).

1. if \( X \perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \) then
2. return \( X \) is a definite non-cause of \( Y \),
3. end if
4. if \( X \not\perp\!\!\!\!\not\perp Y \mid \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \) then
5. return \( X \) is an explicit cause of \( Y \),
6. end if
7. \( \mathcal{M} = \) the set of maximal cliques of \( \text{sib}(X, G^*) \),
8. if exists \( M \in \mathcal{M} \) such that \( X \perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \cup M \), then
9. return \( X \) is a possible cause of \( Y \),
10. end if
11. return \( X \) is an implicit cause of \( Y \).
The complexity of Algorithm 1 can be measured by the maximum number of conditional independence tests (or d-separation queries). Clearly, the maximum number of conditional independence tests performed by Algorithm 1 is \( m + 2 \), where \( m \) is the number of maximal cliques of \( \text{sib}(X, G^*) \). Fortunately, there are only linearly many maximal cliques (with respect to the number of vertices) in a chordal graph (Rose and Tarjan, 1975; Blair and Peyton, 1993), so the number of conditional independence tests needed in Algorithm 1 is at most \( O(|\text{sib}(X, G^*)|) \).

We also present a global learning algorithm as well as its complexity analysis in B.1. The global learning algorithm uses the graphical criteria as well as the properties of different causal relations to identify them. We refer the interested readers to Algorithm 2 and Algorithm 3 in B.1 for more details.

On the other hand, if we use the causal-effect-based method mentioned in Section 1 (or more specifically, use a possible implementation described in B.2 and tested in Section 6), we have to enumerate all possible causal effects of \( X \) on \( Y \) and test whether they are all zeros (non-zeros). In the worst case, the number of conditional independence tests required in this method is \( 2^{|\text{sib}(X, G^*)|} \). Hence, Algorithm 1 is theoretically more efficient than the current causal-effect-based method as the latter needs to perform exponentially many conditional independence tests in the worst case while the former only needs linearly many tests.

6 Experiments

In this section, we illustrate and evaluate the proposed method experimentally using synthetic data sets generated from linear structural equation models with Erdős-Rényi random DAGs. We compare our local ITC with the global one as well as two causal-effect-based methods (CE-based for short): regular testing method (CE-test for short) and multiple testing method (CE-multi for short). In both CE-based methods, we used IDA (Maathuis et al., 2009) to estimate all possible causal effects of a treatment on a target and checked whether all of these causal effects are zeros. The CE-test method checks each effect one by one and the CE-multi tests all effects simultaneously with multiple testing methods.

In Section 6.1, we assume that the true CPDAG or its local structure of interest is available. In this case, the synthetic data was only used to estimate causal effects by the CE-based methods, and to perform conditional independence tests by the local ITC method. Therefore, in these experiments,
we can compare our proposed methods with CE-based methods directly since structure learning from data is not needed and no corresponding error is introduced. In Section 6.2 we further evaluate the methods without providing the true CPDAG or its local structure. Three global structure learning algorithms, including the PC algorithm (Spirtes and Glymour, 1991), the stable PC algorithm (Colombo and Maathuis, 2014) and the GES algorithm (Chickering, 2002a), were used to learn CPDAGs, and the variant of MB-by-MB (Liu et al., 2020b) was used to learn parents and siblings of the vertices of interest. In all of these experiments, algorithms like PC, stable PC, GES, and IDA were called from \texttt{pcalg R-package} (Kalisch et al., 2012), significance level \( \alpha \) of all statistical independence tests was set to be 0.001\(^3\) and all codes were run on a computer with an Intel 2.5GHz CPU and 8 GB of memory.

Let \( \text{ER}(n,d) \) denote a random DAG with \( n \) vertices and average in-and-out degree \( d \). In our experiment, \( n \) is chosen from \{20, 30, 40, 50, 70, 100\} and \( d \) is chosen from \{1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}. To generate a data set, we first sampled an \( \text{ER}(n,d) \) random DAG, and for each directed edge \( X_i \to X_j \) in this DAG, we drew an edge weight \( \beta_{ij} \) from a Uniform([0,1.6]) or a Uniform([-1.6, -0.8] \cup [0.8, 1.6]) distribution. Then, we constructed a linear structural equation model as follows,

\[
X_j = \sum_{X_i \in \text{pa}(X_j)} \beta_{ij} X_i + \epsilon_j, \quad j = 1, ..., n, \tag{1}
\]

where \( \epsilon_1, ..., \epsilon_n \) are independent \( \mathcal{N}(0,1) \) noises. Finally, we drew \( N \) samples from this linear model. In Section 6.1 we set \( N \in \{50, 250\} \), and in Section 6.2 we set \( N \in \{500, 1000, 3000\} \).

We generated 5000 data sets for each setting, and for each data set we randomly sampled a treatment variable and a target variable. We then explored the causal relation between the treatment and the target and compared it with the true one read from the corresponding CPDAG of the sampled DAG. Since there are three types of causal relations, we use the Kappa coefficient as well as the true positive rate (TPR) and the false positive rate (FPR) to measure the performance of each method.

6.1 Learning with True Graphs

In this section, the true CPDAGs or their local structures are provided to exclude estimation biases caused by graph structure learning from data. Since

\(^3\)Experiments show different significance levels give similar results.
| $N$ | $d$ | Method | Average degree | 2     | 3     | 4     |
|-----|-----|--------|----------------|-------|-------|-------|
|     |     |        |                | 2     | 3     | 4     |
| 50  | 2   | CE-test| 2.9619 (0.9041)| 2.8218 (0.8119)| 2.7001 (0.7926)|
|     |     | CE-multi| 3.0261 (0.9172)| 2.8742 (0.8226)| 2.7451 (0.8002)|
| 250 | 3   | CE-test| 1.8515 (0.4090)| 1.7630 (0.3461)| 1.7049 (0.2990)|
|     |     | CE-multi| 1.8726 (0.4159)| 1.7808 (0.3507)| 1.7211 (0.3005)|

Table 1: The detailed TPRs and FPRs of different methods on 100-node random graphs with $d \in \{2, 3, 4\}$ and positive weights. The true graph structures are provided. The standard deviations are omitted as they are all below 0.002.

| $N$ | $d$ | Method | Average degree | 2     | 3     | 4     |
|-----|-----|--------|----------------|-------|-------|-------|
|     |     |        |                | 2     | 3     | 4     |
| 50  | 2   | CE-test| 0.9992 (0.0354)| 0.6211 (0.0004)| 0.6984 (0.0008)|
|     |     | CE-multi| 0.9899 (0.4937)| 0.3684 (0.0016)| 0.3016 (0.0013)|
| 50  | 3   | local ITC | 0.9996 (0.5130)| 0.4216 (0.0031)| 0.4778 (0.0004)|
|     |     | CE-test | 0.9864 (0.6302)| 0.3088 (0.0056)| 0.1889 (0.0168)|
|     |     | CE-multi| 0.9870 (0.6484)| 0.2892 (0.0048)| 0.1778 (0.0168)|
| 250 | 3   | local ITC | 0.9990 (0.0843)| 0.8879 (0.0012)| 0.8800 (0.0008)|
|     |     | CE-test | 0.9913 (0.0904)| 0.6293 (0.0018)| 0.8400 (0.0139)|
|     |     | CE-multi| 0.9921 (0.1024)| 0.6207 (0.0016)| 0.8000 (0.0135)|
| 250 | 4   | local ITC | 0.9983 (0.2077)| 0.7937 (0.0037)| 0.6959 (0.0008)|
|     |     | CE-test | 0.9828 (0.2404)| 0.6667 (0.0052)| 0.6014 (0.0198)|
|     |     | CE-multi| 0.9839 (0.2463)| 0.6614 (0.0042)| 0.5946 (0.0198)|
|     |     |        |                | 0.9989 (0.2789)| 0.7647 (0.0100)| 0.5587 (0.0009)|
|     |     |        |                | 0.9767 (0.3281)| 0.6606 (0.0128)| 0.4441 (0.0267)|
|     |     |        |                | 0.9770 (0.3351)| 0.6561 (0.0140)| 0.4155 (0.0267)|

Table 2: The averages and standard deviations (in parentheses) of ratios of the time used by CE-based methods to local ITC on 100-node random graphs with $d \in \{2, 3, 4\}$ and positive weights. The true graph structures are provided.

the output of the global ITC is invariant when the true CPDAG is provided, the TPR and FPR for learning each type of causal relation are 1 and 0 respectively. Except the global ITC, the local ITC and CE-based methods
**Table 3:** The detailed TPRs and FPRs of different methods on 100-node random graphs with $d \in \{2, 3, 4\}$ and mixed weights. The true graph structures are provided. The standard deviations are omitted as they are all below 0.002.

| $N$ | $d$ | Method     | Def. non-causes | Poss. causes | Def. causes |
|-----|-----|------------|-----------------|-------------|-------------|
|     |     |            | TPR FPR         | TPR FPR     | TPR FPR     |
| 50  | 2   | local ITC  | 0.9994 0.5455   | 0.4167 0.0006 | 0.4828 0.0004 |
|     |     | CE-test    | 0.9957 0.7532   | 0.1979 0.0008 | 0.0862 0.0063 |
|     |     | CE-multi   | 0.9959 0.7792   | 0.1771 0.0002 | 0.0862 0.0063 |
| 50  | 3   | local ITC  | 0.9991 0.6359   | 0.3263 0.0006 | 0.3892 0.0008 |
|     |     | CE-test    | 0.9938 0.8291   | 0.1368 0.0010 | 0.0838 0.0093 |
|     |     | CE-multi   | 0.9938 0.8291   | 0.1368 0.0010 | 0.0838 0.0093 |
| 50  | 4   | local ITC  | 0.9989 0.7559   | 0.1509 0.0034 | 0.2648 0.0009 |
|     |     | CE-test    | 0.9917 0.8770   | 0.0862 0.0036 | 0.0841 0.0088 |
|     |     | CE-multi   | 0.9917 0.8843   | 0.0776 0.0036 | 0.0779 0.0088 |

**Table 4:** The averages and standard deviations (in parentheses) of ratios of the time used by CE-based methods to local ITC on 100-node random graphs with $d \in \{2, 3, 4\}$ and mixed weights. The true graph structures are provided.

| $N$ | Method | Average degree |
|-----|--------|----------------|
|     |        | 2             | 3             | 4             |
| 50  | CE-test| 2.9696 (0.9242) | 2.8080 (0.8331) | 2.7102 (0.7684) |
|     | CE-multi| 3.0357 (0.9368) | 2.8487 (0.8365) | 2.7525 (0.7940) |
| 250 | CE-test| 1.8377 (0.3810) | 1.7797 (0.3570) | 1.7242 (0.2959) |
|     | CE-multi| 1.8623 (0.3935) | 1.8004 (0.3608) | 1.7380 (0.3040) |

need to perform statistical tests. To assess these methods, we run experiments on data with positive weights (Uniform([0.8, 1.6])), as well as a mixture of negative and positive weights (Uniform([-1.6, -0.8] ∪ [0.8, 1.6])), which
generally increases the chance of violations of the faithfulness assumption. We report the averages of TPR, FPR in Table 1 and Table 3, and the averages and standard deviations of ratios of the time used by the CE-based methods to the local ITC in Table 2 and Table 4. In these tables, we use ‘def.’ and ‘poss.’ as abbreviations for ‘definite’ and ‘possible’ respectively.

We can see that the proposed local ITC is significantly better than the CE-based methods in terms of accuracy in Table 1 and Table 3, especially when learning possible and definite causes. Moreover, the time consumed by the local ITC is much less than that consumed by the CE-based methods, as shown in Table 2 and 4. Besides, the sample size $N$ and the degree $d$ have similar effects on all methods: the larger the sample size and the smaller the average degree, the better their performance.

Benefiting from less conditional independence tests, the local ITC algorithm is more stable, more accurate, and more efficient than the CE-based methods, especially when identifying definite causal relations. Comparing Table 3 with Table 1, one can see that although all TPRs drop, the TPRs of the local ITC drop less than those of the CE-based methods. For instance, in Table 1, for the case of $N = 250$, the TPRs of the local ITC and CE-test for identifying definite causal relations are 0.8800 and 0.8400 respectively, while when negative weights are allowed in Table 3, the TPRs of the local ITC and CE-test decrease to 0.7183 and 0.4366, equivalent to 18% and 48% reductions, respectively.

### 6.2 Learning with Estimated Graphs

In this section, we further study experimentally our proposed methods when the true causal structures are not available. We used the variant of MB-by-MB (Liu et al., 2020b) to learn parents and siblings of the vertices of interest, and used the PC algorithm, the stable PC algorithm and GES to learn entire CPDAGs.

Figure 3 shows the Kappa coefficients of different methods based on 20-, 50- and 100-node graphs. As one can see from Figure 3, the proposed local ITC outperforms the other methods at almost all settings. The global ITC combined with PC or PCS is also competitive when the graph is extremely sparse ($d < 2$) or relatively dense ($d > 3.5$). This is probably due to the fact that PC and PCS can effectively learn the sparse graph structures but none of the considered structure learning methods performs well on dense graphs. Moreover, the ITC methods are generally better than the CE-based methods, since the CE-based methods and its local versions perform more additional statistical tests and thus introduce more errors than the global
Figure 3: The Kappa coefficients of different methods on random graphs with positive weights. The graph structures are learned from data.

In Table 5, we report the TPRs and FPRs of different methods for identifying each type of causal relation based on 100-node graphs with average degree $d \in \{2, 3, 4\}$. The sample size $N$ is set to 1000. Table 5 shows that the local ITC is more stable than others, especially on relatively sparse graphs. It can be seen that when $d = 2$ and $d = 3$, the FPR of the local ITC is always one of the best three FPRs, and the TPR of the local ITC is one of the highest three TPRs except for learning non-causal relations when
Table 5: The detailed TPRs and FPRs of different methods on 100-node random graphs. The with sample size $N = 1000$, average degree $d \in \{2, 3, 4\}$ and positive weights. The standard deviations are omitted as they are all below 0.002.

$d = 3$. Moreover, the local ITC performs considerably well when learning possible causal relations and definite causal relations, and always achieves a relatively low FPR when learning definite non-causal relations.

We now compare the total computational time of different methods in Figure 4. The total computational time consists of two parts: the time for learning the required graph structure, and the time for identifying the type of causal relation. Generally, the total time is dominated by the learning of graph structure. As shown in Figure 4, since learning a local structure consumes less time than learning a global structure, the local ITC and the local versions of the CE-based methods are more efficient than global ones. Moreover, the global ITC is much faster than the global CE-based methods,
as the latter need perform additional independence tests. In our experiments, regardless of the time used to learn graph structure, the global ITC takes less than 0.001 seconds to identify types, while the global CE-based methods are 10 times slower. This also explains why the blue, green, or purple dashed/dotted lines in Figure 4 are above the corresponding solid lines.

Since local ITC and local CE-based methods use the same local learning algorithm (the variant of MB-by-MB) and the local structure learning dominates the computational time of these methods, the total time of these three
Table 6: The averages and standard deviations (in parentheses) of the ratios of the time used by local CE-based methods to local ITC on 100-node random graphs with positive weights. The graph structures are learned from data.

| N    | Method       | Average degree |
|------|--------------|----------------|
|      |              | 2              | 3             | 4             |
| 500  | CE-test (local) | 1.3735 (0.1798) | 1.3370 (0.1606) | 1.3076 (0.1398) |
|      | CE-multi (local) | 1.3830 (0.1811) | 1.3434 (0.1635) | 1.3148 (0.1407) |
| 1000 | CE-test (local) | 1.2998 (0.1446) | 1.2703 (0.1214) | 1.2520 (0.1303) |
|      | CE-multi (local) | 1.3076 (0.1435) | 1.2787 (0.1236) | 1.2617 (0.1308) |
| 3000 | CE-test (local) | 1.1854 (0.0885) | 1.1818 (0.1112) | 1.1762 (0.1190) |
|      | CE-multi (local) | 1.1849 (0.0869) | 1.1829 (0.1078) | 1.1753 (0.1135) |

To compare the computational time of the local ITC and the local versions of the CE-based methods in detail, Table 6 reports the averages and standard deviations of the ratios of CPU time of local CE-based methods to local ITC. It can be seen that the local ITC is faster than the local CE-based methods since all ratios are greater than 1.

7 Concluding Remarks

In this paper, we present a local method for identifying the type of causal relation without evaluating causal effects and learning a global causal structure. In our work, we investigate the existence of a causal path from the cause to the target based on a CPDAG and provide a sufficient and necessary graphical condition to check the existence of a causal path. We also study the graphical properties of each type of causal relation. Inspired by these properties, we further propose a local identification criterion for each type of causal relation, which depends only on the induced subgraph of the true CPDAG over the adjacent variables of the treatment as well as some queries about d-separation relations. The local criteria naturally lead to a local learning algorithm for identifying the type of causal relation if one assumes the faithfulness condition holds. Simulation studies empirically prove that the proposed local algorithm performs well.

Our work introduces the local characterizations of types of causal relations, which are helpful for understanding causal relations hidden behind observational data. Except for the theoretical contributions, our results have many potential applications as well. Firstly, it can guide researchers to
perform interventional studies. For example, no invention is needed if the treatment is a definite non-cause of the target. Secondly, it can be combined with the IDA algorithm to estimate possible causal effects to reduce the computational costs. For instance, if the treatment is a non-cause of the target, then without any computation we can conclude that all possible effects are zero (Maathuis et al., 2009). Besides, our results can be used to decide the significance of estimated causal effects. If the treatment is a definite cause of the target, then all the estimated causal effects, no matter how small they are, are significant. To some extent, our results can make up the shortage of using statistical tests when testing the significance of a possible causal effect. Thirdly, our theorems and algorithms can be used to check whether a variable is a mediator in between a cause and a target. All these applications are important in causal inference.

Our results can be easily extended to interventional essential graphs (He and Geng, 2008; Hauser and Bühlmann, 2012), which can be used to represent Markov equivalence classes where some variables are intervened. Basically, interventional essential graphs are also chain graphs and can be learned from the mixture of observational and interventional data. Extending our proposed concepts, theorems, and algorithms to interventional essential graphs is straightforward. A possible future work is to extend the global characterization for definite causal relations to maximal PDAGs. Maximal PDAGs are generalizations of CPDAGs, and have been frequently used for representing causal background knowledge (Perković et al., 2017; Fang and He, 2020; Perković, 2020; Witte et al., 2020; Guo and Perković, 2020). Another interesting direction is to take hidden variables and selection biases into account. For example, one may extend the results to partially ancestral graphs (Richardson and Spirtes, 2002; Ali et al., 2005; Zhang, 2008).

Acknowledgements

This work was supported by National Key R&D Program of China (2018YFB1004300), NSFC (11671020, 11771028, 11971040).
Appendix

A Graph Terminology

A graph $G$ is defined as a vertex set (or node set) $V$ and an edge set $E$. A graph is directed (undirected, partially directed) if all edges in the graph are directed (undirected, a mixture of directed and undirected). The skeleton of a graph $G$ is an undirected graph resulted from turning every directed edge in $G$ into an undirected edge. Given a subset $V'$ of $V$, the induced subgraph of $G$ over $V'$ is defined as $G' = (V', E')$ where $E' \subseteq E$ contains only edges between vertices in $V'$. If a directed edge $X_i \rightarrow X_j$ occurs in $G$, we call $X_i$ a parent of $X_j$ and $X_j$ a child of $X_i$. Two distinct vertices $X_i$ and $X_j$ are siblings of each other if the undirected edge $X_i - X_j$ appears in $G$. If for any $V' \subseteq V$, there exist $X' \in V'$ and $X \in V \setminus V'$ such that $X$ and $X'$ are adjacent, then the graph is called connected, otherwise, it is disconnected. Furthermore, if there is an edge between any two vertices, then the graph is called complete.

A path is a sequence of distinct vertices $(X_{k_1}, \ldots, X_{k_j})$ such that $X_{k_i}$ is adjacent to $X_{k_{i+1}}$. $X_{k_1}$ and $X_{k_j}$ are endpoints of the path, while other vertices on the path are intermediate vertices (nodes). The length of a path is the number of vertices on the path minus one. A path is called partially directed from $X_{k_1}$ to $X_{k_j}$ if $X_{k_i} \leftarrow X_{k_{i+1}}$ does not occur in $G$ for any $i = 1, \ldots, j - 1$. A partially directed path is directed (undirected) if all edges on the path are directed (undirected). A cycle is a path from a vertex to itself. A partially directed (directed, undirected) cycle can be defined similarly. We note that both directed paths (cycles) and undirected path (cycles) are partially directed. A vertex $X_i$ is an ancestor of $X_j$ and $X_j$ is a descendant of $X_i$ if there is a directed path from $X_i$ to $X_j$ or $X_i = X_j$. A chord of a path (cycle) is any edge joining two nonconsecutive vertices on the path (cycle). A path (cycle) without any chord is called chordless. Any path with length one is chordless. An undirected graph is chordal if it has no chordless cycle with length greater than three. Given a chordal graph $C = (V, E)$, if the induced subgraph of $C$ over $V' \subseteq V$ is complete, then $V'$ is called a clique of $C$. Moreover, if there is no $V''$ such that $V' \subseteq V''$ and $V''$ is a clique, then $V'$ is called a maximal clique. A directed graph is acyclic (DAG) if there are no directed cycles.
B Algorithms

We now provide global ITC and causal-effect-based methods for learning the types of causal relations.

B.1 A Global Learning Algorithm

Given a target variable and a CPDAG, as shown in Example 2, identifying a definite non-causal relation or an explicit causal relation is straightforward. To discriminate an implicit causal relation from a possible causal relation, we need an approach to find critical sets. The next proposition is particularly useful.

Proposition 4. For any two distinct vertices $X, Y$ in a CPDAG $G^*$ such that $X$ is not an explicit cause of $Y$, it holds that $C_{XY} = \bigcup_{Z \in Z} C_{XZ}$, where $C_{UV}$ denotes the critical set of $U$ with respect to $V$ and $Z$ is the set of explicit causes of $Y$ which are also in the chain component containing $X$.

Proposition 4 provides a factorization of the critical set of $X$ with respect to $Y$. For simplicity, we call $\bigcup_{Z \in Z} C_{XZ}$ the critical set of $X$ with respect to $Z$. Algorithm 2 shows how to find $\bigcup_{Z \in Z} C_{XZ}$ efficiently. Algorithm 2 runs a breadth-first-search and returns the critical set of $X$ with respect to $Z$ in $G^*_u$. In Algorithm 2 we start from the siblings of $X$, then search chordless paths from the siblings until reaching some $Z_i \in Z$. Every chordless path starting from a sibling of $X$ is recorded in a queue $S$ as a triple like $(\alpha, \psi, \tau)$, where $\alpha$ and $\tau$ are the start and the end point of the path, respectively, and $\psi$ is the sibling of $\tau$ on the path. If $\tau$ is a member of $Z$, we add $\alpha$ to the critical set $C$ and remove from $S$ all triples where the first element is $\alpha$, that is, we stop enumerating chordless paths starting with $\alpha$. Otherwise, we extend the chordless path to the siblings of $\tau$ that are neither $\psi$ nor siblings of $\psi$ and add the corresponding triples to the queue $S$. In this algorithm, a set of visited triples, $H$, is introduced to speed up the search by avoiding visiting the same triple twice.

Finally, we present a global learning approach for identifying the type of causal relation in Algorithm 3. Algorithm 3 is global in the sense that it takes an entire CPDAG as input. In Algorithm 3 we first check whether $X$ and $Y$ are in the same chain component. If they are, $X$ is a possible cause of $Y$ based on Proposition 2. Otherwise, we find the set of explicit causes of $Y$ and denote it by $Z$. This can be done by searching for the vertices that are connected to $Y$ in the directed subgraph of $G^*$. If $X \in Z$, $X$ is an explicit cause of $Y$, otherwise, we find the critical set $C$ of $X$ with respect to
Algorithm 2 Finding the critical set of a given $X$ with respect to a set $Z$

**Require:** A chordal graph $G_u^*$, a variable $X$ in $G_u^*$, and a variable set $Z \neq \emptyset$ such that $X \notin Z$.

**Ensure:** $C$, which is the critical set of $X$ with respect to $Z$ in $G_u^*$.

1. Initialize $C = \emptyset$, a waiting queue $S = []$, and a set $H = \emptyset$.
2. for $\alpha \in \text{adj}(X)$ do
3. add $(\alpha, X, \alpha)$ to the end of $S$,
4. end for
5. while $S$ is not empty do
6. take the first element $(\alpha, \psi, \tau)$ out of $S$ and add it to $H$,
7. if $\tau \in Z$ then
8. add $\alpha$ to $C$, and remove from $S$ all triples where the first element is $\alpha$,
9. else
10. for $\beta \in \text{adj}(\tau)$ and $\beta \notin \text{adj}(\psi) \cup \{\psi\}$ do
11. if $(\alpha, \tau, \beta) \notin H$ and $(\alpha, \tau, \beta) \notin S$ then
12. add $(\alpha, \tau, \beta)$ to the end of $S$,
13. end if
14. end for
15. end if
16. end while
17. return $C$

$Z$. When $C = \emptyset$, we have that there are no explicit causes of $Y$ in the chain component containing $X$, so $X$ is not a cause of $Y$. Finally, using Theorem 4, Algorithm 3 distinguishes between possible causes and implicit causes.

Since Algorithm 2 does not visit the same triple like $(\alpha, \psi, \tau)$ twice, where $\alpha$ is a sibling of $X$ and $\tau$ is sibling of $\psi$ in $G_u^*$, the complexity of Algorithm 2 in the worst case is $O(|\text{sib}(X, G^*)| \cdot |\text{E}(G_u^*)|)$, where $|\text{E}(G_u^*)|$ is the number of edges in $G_u^*$. Now we consider the computational complexity of global ITC (Algorithm 3). We know that the complexity to check the undirected connectivity of $X$ and $Y$ or to find $\text{an}(Y, G^*)$ is $O(|\text{E}(G^*)|^2)$, where $|\text{E}(G^*)|$ is the number of vertices in $G^*$. Consequently, the complexity of global ITC is $O(|\text{E}(G^*)|^2 + |\text{sib}(X, G^*)| \cdot |\text{E}(G_u^*)|)$. Clearly, the worst case is $O(|\text{E}(G^*)|^3)$.

B.2 Causal-Effect-Based Methods

Section 6 briefly describes the causal-effect-based methods used in our experiments. Now we summarize the detailed procedure in Algorithm 4.
Algorithm 3 A global algorithm for identifying the type of causal relation (global ITC).

Require: A CPDAG $\mathcal{G}^*$, a variable $X$ and a target $Y$ in $\mathcal{G}^*$.

Ensure: The type of causal relation between $X$ and $Y$.

1: if $X$ and $Y$ are connected by a path in $\mathcal{G}^*_u$ then
2: return $X$ is a possible cause of $Y$,
3: end if
4: let $Z = an(Y, \mathcal{G}^*)$,
5: if $X \in Z$ then
6: return $X$ is an explicit cause of $Y$,
7: end if
8: use Algorithm 2 to find the critical set $C$ of $X$ with respect to $Z$ in $\mathcal{G}^*_u$,
9: if $|C| = 0$ then
10: return $X$ is a definite non-cause of $Y$,
11: end if
12: if $C$ induces a complete subgraph of $\mathcal{G}^*_u$ then
13: return $X$ is a possible cause of $Y$,
14: end if
15: return $X$ is an implicit cause of $Y$.

The first four lines in Algorithm 4 are borrowed from the IDA algorithm (Maathuis et al., 2009), which enumerate all possible parental sets of the treatment $X$ and estimate all possible causal effects of $X$ on $Y$. The possible effects are stored in a set, denoted by $\Theta_X$. Next, Algorithm 4 uses regular testing method or multiple testing method to test whether every causal effect in $\Theta_X$ is zero. Based on the testing results, lines 6-12 return the type of causal relation between $X$ and $Y$ by the definitions of different types of causes.

We note that, the input of Algorithm 4 includes a CPDAG $\mathcal{G}^*$ or its induced subgraph over $pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*)$. In the original version of the IDA algorithm, the authors used an entire CPDAG as an input (Maathuis et al., 2009). This is probably due to the fact that there was no efficient approach to learn the induced subgraph over $pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*)$ locally at that time. With the help of the variant of MB-by-MB (Liu et al., 2020b, Algorithm 3), we can now learn the induced subgraph over $pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*)$ efficiently. Therefore, Algorithm 4 could also be local if it is combined with the variant of MB-by-MB.
Algorithm 4 A causal-effect-based algorithm for identifying the type of causal relation.

Require: A treatment $X$, a target $Y$, a CPDAG $\mathcal{G}^*$ or its induced subgraph over $pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*)$.

Ensure: The type of causal relation between $X$ and $Y$.

1: set $\Theta_X = \emptyset$,
2: for each $S \subset sib(X, \mathcal{G}^*)$ such that orienting $S \to X$ and $X \to sib(X, \mathcal{G}^*) \setminus S$ does not introduce any v-structure collided on $X$ do
3: estimate the causal effect of $X$ on $Y$ by adjusting for $S \cup pa(X, \mathcal{G}^*)$,
4: and add the causal effect to $\Theta_X$,
5: end for
6: test whether every causal effect in $\Theta_X$ is zero,
7: if every causal effect in $\Theta_X$ is zero then
8: return $X$ is a definite non-cause of $Y$,
9: end if
10: if every causal effect in $\Theta_X$ is non-zero then
11: return $X$ is a definite cause of $Y$,
12: end if
13: return $X$ is a possible cause of $Y$.

C Detailed Proofs

The proofs of lemmas, theorems and corollaries in the main text of this paper will be presented in this section. Before that, we first introduce some prerequisite concepts and results.

Let $\pi = (v_0, v_1, ..., v_k)$ denote a path with length $k$. The subpath $\pi(v_i, v_j)$ of $\pi$, with $j > i$, is the path $(v_i, v_{i+1}, ..., v_{j-1}, v_j)$. If $k \geq 2$, we say three consecutive vertices $v_i, v_{i+1}$ and $v_{i+2}$ form a triangle on $\pi$ if $v_i$ is adjacent to $v_{i+2}$. $\pi$ is called triangle-free if it does not contain any triangle. For a path in a chordal graph, we find the following result.

**Lemma 3.** In any chordal graph, a path is chordless if and only if it is triangle-free.

**Proof.** Let $\pi = (v_0, v_1, ..., v_k)$ denote a path with length $k \geq 2$. If $\pi$ is chordless, then it is obviously triangle-free. Suppose $\pi$ is not chordless, then we can choose a chord $v_i - v_j$ such that the subpath $\pi(v_i, v_j)$ has no chord except for $v_i - v_j$. If $j = i + 2$, then $v_i, v_{i+1}$ and $v_j$ form a triangle. If $j > i + 2$, then $\pi(v_i, v_j)$ and $v_i - v_j$ form a cycle with length greater than 3. However, since the graph is chordal, we must have a chord $v_k - v_l$ with $i \leq k, l \leq j$.
and \( l \geq k + 2 \) and \( l - k < j - i \). This is contrary to our assumption. \( \square \)

Lemma 3 is useful for finding chordless path, since checking whether a path is triangle-free is much easier. The following is another useful result for chordal graphs.

**Lemma 4.** Let \( \rho \) be a cycle with length greater than 3 in a given chordal graph, and \( X \) be a vertex on \( \rho \). If the two vertices adjacent to \( X \) on \( \rho \) are not adjacent to each other, then \( \rho \) has a chord where \( X \) is an endpoint.

**Proof.** Let \( v_1 \) and \( v_2 \) be two vertices adjacent to \( X \) on \( \rho \). Suppose that \( \rho \) does not have a chord where \( X \) is an endpoint. Since \( \rho \) has length greater than 3, \( \rho \) must have a chord. Clearly, any chord of \( \rho \) separates \( \rho \) into two sub-cycles. By assumption, it is easy to check that at least one sub-cycle contains \( X \), \( v_1 \) and \( v_2 \). If this sub-cycle still has a chord, then we can construct another cycle containing \( X \), \( v_1 \) and \( v_2 \) but with shorter length. Finally, we will have a cycle containing \( X \), \( v_1 \) and \( v_2 \) without any chord. Since \( v_1 \) and \( v_2 \) are not adjacent, the length of this cycle must be greater than 3, which is contradicted to the definition of chordal graph. \( \square \)

A chordal graph \( C \) can be turned into a directed graph by orienting its edges. If the resulting directed graph is a DAG without v-structure, then these orientations form a v-structure-free acyclic orientation of \( C \) (Bernstein and Tetali, 2017). Any v-structure-free acyclic orientation of a connected chordal graph has a unique source, that is, a vertex which has no parent. Conversely, any vertex in a connected chordal graph can be the unique source in some v-structure-free acyclic orientation (Blair and Peyton, 1993; Bernstein and Tetali, 2017). Recall that the undirected subgraph of a CPDAG is the union of disjoint connected chordal graphs called chain components (Andersson et al., 1997). Maathuis et al. (2009) argued that any v-structure-free acyclic orientation of the edges in \( G^* \) corresponds to a DAG in the equivalence class represented by \( G^* \), and such an orientation can be considered separately for each of the disjoint chordal graphs (or chain components). Moreover, Maathuis et al. (2009) proved that,

**Lemma 5.** (Maathuis et al. 2009, Lemma 3.1) Let \( G^* \) be a CPDAG, \( X \) be a vertex of \( G^* \), and \( S \subseteq \text{ne}(X, G^*) \). Then there is a DAG \( G \in [G^*] \) such that \( \text{pa}(X, G) = \text{pa}(X, G^*) \cup S \) if and only if orienting \( S \rightarrow X \) and \( X \rightarrow D \) for every \( S \in S \) and \( D \in \text{sib}(X, G^*) \setminus S \) in \( G^* \) does not introduce any new v-structure.

28
Meek (1995, Lemma 1) proved that if \( Y \in \text{pa}(X, G^*) \), then \( Y \in \text{pa}(X', G^*) \) for every \( X' \in \text{ne}(X, G^*) \). From this result we can prove that the condition in Lemma 5 holds if and only if \( S \) is a clique. As we will see, Lemma 5 plays a key role in proving the main results of this paper, as it provides a simple and local criterion for checking whether a subset of \( X \)'s siblings can be \( X \)'s parents in some equivalent DAGs.

Let \( \pi \) denote a path. A subsequence of \( \pi \) is obtained by deleting some vertices from \( \pi \) without changing the order of the remaining vertices. The final prerequisite result is about the relation between directed paths and partially directed paths.

**Lemma 6.** There is a directed path from \( X \) to \( Y \) in \( G^* \) if and only if there is a partially directed path from \( X \) to \( Y \) in \( G^* \) on which the node adjacent to \( X \) is a child of \( X \).

**Proof.** The necessity is trivial. For sufficiency, let \( \pi = (X, v, ..., Y) \) be the partially directed path from \( X \) to \( Y \) in \( G^* \) such that \( X \to v \). Assume that \( w \) is the first vertex from the side of \( Y \) which is adjacent to \( X \), then we have \( X \to w \). Now consider \( \pi(w, Y) \). As \( \pi(w, Y) \) is also partially directed, by Perković et al. (2017, Lemma 3.6), there is a subsequence \( \pi^* \) of \( \pi(w, Y) \) forms a chordless partially directed path from \( X \) to \( Y \) in \( G^* \). Let \( \pi^{**} \) denote the path by concatenating \( X \to w \) and \( \pi^* \), then \( \pi^{**} \) is a partially directed path from \( X \) to \( Y \) on which the node adjacent to \( X \) is a child of \( X \). By construction, \( X \) is not adjacent to any vertex on \( \pi^{**} \) except for \( w \). Thus, by Maathuis and Colombo (2015, Lemma 7.2), \( \pi^{**} \) is a directed path. \( \square \)

In the following Appendices C.1 to C.13, we will present the detailed proofs of the main results provided in the main text, with the help of the aforementioned concepts and lemmas.

**C.1 Proof of Lemma 1**

**Proof.** Given a CPDAG \( G^* \), for any DAG \( G \in [G^*] \), Fang and He (2020, Lemma 2) showed that a variable \( X \) is not a cause of another variable \( Y \) in \( G \) if and only if the critical set of \( X \) with respect to \( Y \) in \( G^* \), which is denoted by \( C \), is a subset of \( \text{pa}(X, G) \). Consequently, \( X \) is a cause of \( Y \) in \( G \) if and only if \( C \) is not a subset of \( \text{pa}(X, G) \). That is, some vertex in \( C \) must be a child of \( X \) in \( G \). The desired result comes from the definition of definite cause. \( \square \)
C.2 Proof of Lemma 2

Proof. We first show the necessity. By the definition, \( C \subseteq \text{sib}(X, G^*) \cup \text{ch}(X, G^*) \). Let \( G \in [G^*] \) be an arbitrary DAG. If \( C \cap \text{ch}(X, G) = \emptyset \) and \( C \neq \emptyset \), then \( C \subseteq \text{pa}(X, G) \), and thus we have \( C \subseteq \text{sib}(X, G^*) \). Maathuis et al. (2009, Lemma 3) proved that a non-empty subset of \( \text{sib}(X, G^*) \) can be a part of \( X \)'s parent set in some equivalent DAG if and only if the subset induces a complete subgraph. Therefore, \( C \) induces a complete subgraph of \( G^* \). This completes the proof of the necessity. We next prove the sufficiency. If \( C = \emptyset \), then it is clear that \( C \cap \text{ch}(X, G) = \emptyset \) for some \( G \in [G^*] \). Now assume \( C \neq \emptyset \) and \( C \) induces a complete subgraph of \( G^* \) and \( C \cap \text{ch}(X, G^*) = \emptyset \). As \( C \subseteq \text{sib}(X, G^*) \cup \text{ch}(X, G^*) \), we have \( C \subseteq \text{sib}(X, G^*) \). Again, by Maathuis et al. (2009, Lemma 3), there is a DAG \( G \) in \( [G^*] \) such that \( C \subseteq \text{pa}(X, G) \). Therefore, \( C \cap \text{ch}(X, G) = \emptyset \). □

C.3 Proof of Theorem 1

Proof. Theorem 1 follows from Lemmas 1 and 2 directly. □

C.4 Proof of Proposition 1

Proof. Denote the CPDAG containing \( X \) and \( Y \) by \( G^* \). It suffices to show that, if \( X \) and \( Y \) are in the same chain component, then there exists a DAG in \( [G^*] \) in which \( Y \) is an ancestor of \( X \). By Lemma 5, there exists a DAG \( G \) in \( [G^*] \) such that \( \text{pa}(Y, G) = \text{pa}(Y, G^*) \) and \( \text{ch}(Y, G) = \text{ch}(Y, G^*) \cup \text{sib}(Y, G^*) \). Let \( \pi = (Y, v_1, ..., X) \) be the shortest path from \( Y \) to \( X \). It is clear that \( \pi \) has no chord. Moreover, the corresponding path of \( \pi \) in \( G^* \) is undirected as \( X \) and \( Y \) are in the same chain component. On the other hand, \( Y \to v_1 \) is in \( G \) by our construction. Hence, according to Perković et al. (2017, Lemma B.1), \( \pi \) is a directed path. □

C.5 Proof of Proposition 2

Proof. According to the definition of partially directed path, an undirected path is also partially directed, hence if \( X \) and \( Y \) are in the same chain component, they are possible causes of each other by Theorem 2 and Proposition 1. Conversely, if \( X \) and \( Y \) are possible causes of each other, then by Theorem 2 there is a partially directed path from \( X \) to \( Y \) as well as a partially directed path from \( Y \) to \( X \). Clearly, neither of these two paths contains a directed edge, otherwise, a partially directed cycle containing directed edges would
occur. Therefore, \(X\) and \(Y\) are connected by an undirected path, which means they are in the same chain component.

\[\square\]

**C.6 Proof of Proposition 3**

**Proof.** Let \(Z\) be a vertex in the chain component containing \(X\), then every partially directed path between \(Z\) and \(Y\), if any, must pass through \(X\). Since there is a v-structure-free orientation of the chain component whose unique source is \(X\), there is a DAG in the Markov equivalence class represented by \(G^*\) such that none of the vertex in the chain component is an ancestor of \(Y\) except \(X\).

\[\square\]

**C.7 Proof of Proposition 4**

**Proof.** If \(X\) and \(Y\) are in the same chain component, then \(Z = \{Y\}\) and the equation trivially holds. Suppose that \(X\) and \(Y\) are not in the same chain component. We first prove that \(C_{XY} \subseteq \cup_{Z \in Z} C_{XZ}\). Without loss of generality, we can assume \(C_{XY} \neq \emptyset\). By the definition of critical set, for any \(C \in C_{XY}\), there is a chordless partially directed path \(\rho\) from \(X\) to \(Y\) on which \(C\) is adjacent \(X\). Since \(X\) and \(Y\) are not in the same chain component, \(\rho\) must contain a directed edge. Let \(Z\) be the vertex on \(\rho\) such that \(\rho(Z,Y)\) starts with a directed edge and \(Z\) is in the chain component containing \(X\). By Maathuis and Colombo (2015, Lemma 7.2) or Perković et al. (2017, Lemma B.1), \(\rho(Z,Y)\) is a directed path. Therefore, \(Z\) is an explicit cause of \(Y\). Since \(X\) is not an explicit cause of \(Y\), we have \(Z \neq X\), and thus \(\rho(X,Z)\) is a chordless undirected path. This means \(C \in C_{XZ}\). As \(C \in C_{XY}\) is arbitrary, we have \(C_{XY} \subseteq \cup_{Z \in Z} C_{XZ}\). Conversely, for any \(Z \in Z\) and \(C \in C_{XZ}\), there is a chordless undirected path \(\pi_1\) from \(X\) to \(Z\) on which \(C\) is adjacent \(X\). Let \(\pi_2\) be the shortest directed path from \(Z\) to \(Y\). As \(X\) and \(Y\) are not in the same chain component, \(Z \neq Y\). Hence, concatenating \(\pi_1\) and \(\pi_2\) results a partially directed path from \(X\) to \(Y\) with length greater than 1. Denote such a path by \(\pi\). If \(\pi\) is chordless, then we have \(C \in C_{XY}\). If this is not the case, then \(\pi\) must have a chord connecting one vertex \(v_1\) on \(\pi_1\) and another vertex \(v_2\) on \(\pi_2\). Clearly, the edge between \(v_1\) and \(v_2\) should be directed, and the direction is \(v_1 \rightarrow v_2\). Since \(X\) is not an explicit cause of \(Y\), it holds that \(v_1 \neq X\). With out loss of generality, we assume \(v_1\) is the first vertex from \(X\’s\) side who are adjacent to some \(v_2\) on \(\pi_2\), then concatenating \(\pi(X,v_1), v_1 \rightarrow v_2\) and \(\pi(v_2,Y)\) results another partially directed path \(\pi'\) which is shorter than \(\pi\). It is easy to verify that \(\pi'\) is chordless, and \(C\) is still adjacent to \(X\) on \(\pi'\). Therefore, \(C \in C_{XY}\), and consequently we have
\[ \cup_{v \in \mathcal{Z}} C_{XZ} \subseteq C_{XY}. \] This completes the proof of Proposition 4. \qed

C.8 Proof of Theorem 2

Proof. Suppose \( X \) is a definite non-cause of \( Y \), then for every DAG \( \mathcal{G} \) in the Markov equivalence class represented by \( \mathcal{G}^* \), \( Y \) is a non-descendant of \( X \). Since Lemma 5 indicates that there is a DAG \( \mathcal{G} \) such that \( pa(X, \mathcal{G}) = pa(X, \mathcal{G}^*) \) and \( ch(X, \mathcal{G}) = adj(X, \mathcal{G}^*) \setminus pa(X, \mathcal{G}^*) \), we have \( X \perp \! \! \! \! \perp Y \mid pa(X, \mathcal{G}^*) \) by local Markov property. On the other hand, if \( X \) is a definite cause or a possible cause of \( Y \), then by definition there is a DAG \( \mathcal{G} \) in the Markov equivalence class represented by \( \mathcal{G}^* \) in which \( X \) is an ancestor of \( Y \). Assume \( \pi \) is a directed path from \( X \) to \( Y \) in \( \mathcal{G} \). Since every vertex on \( \pi \) is a non-collider and none of the vertices on \( \pi \) is in \( pa(X, \mathcal{G}^*) \), \( X \not\perp \! \! \! \! \perp Y \mid pa(X, \mathcal{G}^*) \). \qed

C.9 Proof of Theorem 3

Proof. If \( X \) is an explicit cause of \( Y \), then there is a directed path \( \pi \) from \( X \) to \( Y \) in \( \mathcal{G}^* \). Hence, for any DAG \( \mathcal{G} \) in the Markov equivalence class represented by \( \mathcal{G}^* \), \( \pi \) is directed in \( \mathcal{G} \), which means \( \pi \) has no collider in \( \mathcal{G} \). However, none of the vertices on \( \pi \) is a member of \( pa(X, \mathcal{G}^*) \) or \( sib(X, \mathcal{G}^*) \), since otherwise, a directed cycle or a partially directed cycle with directed edges would occur in \( \mathcal{G}^* \). Therefore, \( \pi \) is active given \( pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \), which means \( X \not\perp \! \! \! \! \perp Y \mid pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \). Conversely, suppose \( X \) is not an explicit cause of \( Y \). In the following, we will prove that \( X \not\perp \! \! \! \! \perp Y \mid pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \) holds. By Lemma 5, there is a DAG \( \mathcal{G} \) in the Markov equivalence class represented by \( \mathcal{G}^* \) such that \( ch(X, \mathcal{G}) = sib(X, \mathcal{G}^*) \cup ch(X, \mathcal{G}^*) \) and \( pa(X, \mathcal{G}) = pa(X, \mathcal{G}^*) \). Consider a path \( \pi \) from \( X \) to \( Y \) in \( \mathcal{G} \). If the length of \( \pi \) is 1, then the corresponding path of \( \pi \) in \( \mathcal{G}^* \) must be \( X \leftarrow Y \) or \( X \rightarrow Y \). Thus, \( \pi \) is blocked given \( pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \). If the length of \( \pi \) is greater than 1, without loss of generality we can assume \( \pi = (X, v_1, \ldots, v_n, Y) \). If \( v_1 \in pa(X, \mathcal{G}) \), then \( \pi \) is blocked by \( pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \) since \( v_1 \) cannot be a collider on \( \pi \). If \( v_1 \in ch(X, \mathcal{G}^*) \), then \( \pi \) is not directed, since otherwise, the corresponding path in \( \mathcal{G}^* \) would be a partially directed path from \( X \) to \( Y \) where the node adjacent to \( X \) is a child of \( X \). Therefore, there must be a collider on \( \pi \). Let \( v_i \) be the collider nearest to \( X \). If \( v_i \in an(pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*), \mathcal{G}) \), there exists a partially directed cycle with directed edges in \( \mathcal{G}^* \), which is impossible. Thus, \( v_i \notin an(pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*), \mathcal{G}) \), and \( \pi \) is blocked by \( pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \). Finally, in the case where \( v_i \in sib(X, \mathcal{G}^*) \), if \( v_i \) is a non-collider, \( \pi \) is clearly blocked by \( pa(X, \mathcal{G}^*) \cup sib(X, \mathcal{G}^*) \). If \( v_i \) is a collider, then \( v_2 \) is adjacent to \( X \), which means \( v_2 \notin ch(X, \mathcal{G}^*) \), since otherwise, both
Therefore, with directed edges. This means \( v_2 \in \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \). Since \( v_2 \) is a non-collider on \( \pi \), \( \pi \) is blocked by \( \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \). This completes the proof of Theorem 3.

C.10 Proof of Theorem 4

Proof. Let \( C \) be the critical set of \( X \) with respect to \( Y \) in \( G^* \). Suppose that \( X \) is an implicit cause of \( Y \), then by Theorem 3 \( X \perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \). For any \( M_w \in \mathcal{M} \), from Theorem 1 we know that \( C \setminus M_w \neq \emptyset \). Therefore, according to Proposition 4 there is a partially directed path from \( X \) to \( Y \), denoted by \( \pi_w = (X - w_1 - ... - w_t - Z_w \rightarrow ... \rightarrow Y) \), such that \( X - w_1 - ... - w_t - Z_w \) is chordless and \( w_1 \notin M_w \). Since every partially directed cycle in \( G^* \) is an undirected cycle, none of the vertices on \( \pi_w \) is a parent of \( X \) in \( G^* \). Moreover, due to the chordless-ness, if \( w_1 \neq Z_w \), then none of \( w_2, ..., w_t, Z_w \) is adjacent to \( X \) and thus none of them is in \( M_w \). (If \( w_1 = Z_w \), then it is clear that \( Z_w \notin M_w \).) Since by Lemma 5 there is a DAG in the Markov equivalence class represented by \( G^* \) such that \( \pi_w \) is directed, \( \pi_w \) is active given \( \text{pa}(X, G^*) \cup M \). Therefore, \( X \not\perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \cup M \) for any \( M \in \mathcal{M} \). Conversely, \( X \perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \) implies \( X \) is not an explicit cause of \( Y \), which also means \( Y \notin \text{ch}(X, G^*) \). Moreover, \( X \not\perp \!\!\!\!\perp Y \mid \text{pa}(X, G^*) \cup M \) for any \( M \in \mathcal{M} \) implies \( Y \notin \text{pa}(X, G^*) \cup \text{sib}(X, G^*) \). Therefore, \( X \) and \( Y \) are not adjacent. Suppose that \( X \) is not implicit. Since \( X \) is not an explicit cause of \( Y \), \( C \cap \text{ch}(X, G^*) = \emptyset \). Thus, by Theorem 1 there exists an \( M \in \mathcal{M} \) such that \( C \) is a subset of \( M \). (If \( C = \emptyset \), then for any \( M \in \mathcal{M} \), \( C \subset M \).) We will show that \( \text{pa}(X, G^*) \cup M \) d-separates \( X \) and \( Y \). By Lemma 5 there is a DAG \( G \) in the Markov equivalence class represented by \( G^* \) such that \( \text{ch}(X, G) = \text{sib}(X, G^*) \cup \text{ch}(X, G^*) \setminus M \) and \( \text{pa}(X, G) = \text{pa}(X, G^*) \cup M \). Let \( \pi = (X, v_1, ..., v_n, Y) \) be an arbitrary path connecting \( X \) and \( Y \) in \( G \). The length of \( \pi \) should be greater than 1 as \( X \) and \( Y \) are not adjacent. If \( v_1 \) is a parent of \( X \) in \( G \), then clearly \( \pi \) is blocked by \( \text{pa}(X, G^*) \cup M \), since \( v_1 \) is a non-collider on \( \pi \) and \( v_1 \in \text{pa}(X, G^*) \cup M \) by the construction of \( G \). Now assume that \( v_1 \) is a child of \( X \) in \( G \). If \( v_1 \in \text{ch}(X, G^*) \), then there must be a collider on \( \pi \), since otherwise, the corresponding path of \( \pi \) in \( G^* \) is a partially directed path where the node adjacent to \( X \) is a child of \( X \), which means \( X \) is an explicit cause of \( Y \) according to Lemma 6. Clearly, the collider nearest to \( X \) on \( \pi \) is not an ancestor of \( \text{pa}(X, G^*) \cup M \). Thus, \( \pi \) is blocked by \( \text{pa}(X, G^*) \cup M \). For the same reason, if \( v_1 \in \text{sib}(X, G^*) \setminus M \) and there is a collider on \( \pi \), then \( \pi \) is blocked by \( \text{pa}(X, G^*) \cup M \) due to the fact that the collider nearest to \( X \) on \( \pi \) can not be an ancestor of \( \text{pa}(X, G^*) \cup M \).
Finally, if $v_1 \in \text{sib}(X, G^*) \setminus M$ and there is no collider on $\pi$, then $\pi$ is directed in $G$, and the corresponding path of $\pi$ in $G^*$ is partially directed. Let $Z$ be the vertex on $\pi$ such that the subpath $\pi(X, Z)$ is undirected in $G^*$ and $Z$ is an explicit cause of $Y$. Obviously, such $Z$ exists, and $Z \neq Y$ or $X$. Since $v_1 \notin M$, we have $v_1 \notin C$ and thus $\pi(X, Z)$ has a chord. By Perković et al. (2017 Lemma 3.6), there is a subsequence $\pi^*$ of $\pi(X, Z)$ forms a chordless undirected path from $X$ to $Z$ in $G^*$. Together with Proposition 4, this result indicates that there is a vertex $w$ on $\pi(X, Z)$ such that $w \in C$. However, by construction, $w \in pa(X, G)$, which makes $\pi(X, w)$ and $w \rightarrow X$ form a directed cycle in $G$. Thus, $\pi$ must contain a collider. This completes the proof.

C.11 Proof of Theorem 5

Proof. The proof directly follows from Theorems 1 to 4, as well as Propositions 2 and 4.

References

A. R. Ali, T. S. Richardson, P. Spirtes, and J. Zhang. Towards characterizing Markov equivalence classes for directed acyclic graphs with latent variables. In Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence, pages 10–17. AUAI press, 2005.

S. A. Andersson, D. Madigan, and M. D. Perlman. A characterization of Markov equivalence classes for acyclic digraphs. The Annals of Statistics, 25(2):505–541, 04 1997.

Y. Bengio, T. Deleu, N. Rahaman, N. R. Ke, S. Lachapelle, O. Bilaniuk, A. Goyal, and C. Pal. A meta-transfer objective for learning to disentangle causal mechanisms. In International Conference on Learning Representations, 2020.

M. Bernstein and P. Tetali. On sampling graphical Markov models. arXiv e-prints, art. arXiv:1705.09717, May 2017.

J. R. S. Blair and B. Peyton. An introduction to chordal graphs and clique trees. In Graph Theory and Sparse Matrix Computation, pages 1–29, New York, NY, 1993. Springer New York.

D. M. Chickering. Learning equivalence classes of Bayesian-network structures. Journal of machine learning research, 2(Feb):445–498, 2002a.
D. M. Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3(Nov):507–554, 2002b.

D. Colombo and M. H. Maathuis. Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15:3921–3962, 2014.

Z. Fang and Y. He. IDA with background knowledge. In *Proceedings of the Thirty-sixth Conference on Uncertainty in Artificial Intelligence*. PMLR, 2020.

S. Fu and M. C. Desmarais. Markov blanket based feature selection: a review of past decade. In *Proceedings of the World Congress on Engineering*, volume 1, page 321–328. Newwood Ltd., 2010.

T. Gao and Q. Ji. Local causal discovery of direct causes and effects. In *Advances in Neural Information Processing Systems 28*, pages 2512–2520. Curran Associates, Inc., 2015.

F. R. Guo and E. Perković. Minimal enumeration of all possible total effects in a Markov equivalence class. *arXiv e-prints*, art. arXiv:2010.08611, Oct. 2020.

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

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

Y. He, J. Jia, and B. Yu. Counting and exploring sizes of Markov equivalence classes of directed acyclic graphs. *Journal of Machine Learning Research*, 16:2589–2609, 2015.

M. Kalisch, M. Mächler, D. Colombo, M. H. Maathuis, and P. Bühlmann. Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, 47(11):1–26, 2012.

M. Koivisto and K. Sood. Exact Bayesian structure discovery in Bayesian networks. *Journal of Machine Learning Research*, 5(May):549–573, 2004.

M. J. Kusner, J. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, volume 30, pages 4066–4076. Curran Associates, Inc., 2017.
S. L. Lauritzen and T. S. Richardson. Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):321–348, 2002.

S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 50(2):157–224, 1988.

Y. Liu, Z. Cai, C. Liu, and Z. Geng. Local learning approaches for finding effects of a specified cause and their causal paths. *ACM Transactions on Intelligent Systems and Technology*, 10(5), Sep 2019.

Y. Liu, Z. Fang, Y. He, and Z. Geng. Collapsible IDA: Collapsing parental sets for locally estimating possible causal effects. In *Proceedings of the Thirty-sixth Conference on Uncertainty in Artificial Intelligence*. PMLR, 2020a.

Y. Liu, Z. Fang, Y. He, Z. Geng, and C. Liu. Local causal network learning for finding pairs of total and direct effects. *Journal of Machine Learning Research*, 21(148):1–37, 2020b.

M. H. Maathuis and D. Colombo. A generalized back-door criterion. *The Annals of Statistics*, 43(3):1060–1088, 06 2015.

M. H. Maathuis, M. Kalisch, and P. Bühlmann. Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37(6A):3133–3164, 12 2009.

C. Meek. Causal inference and causal explanation with background knowledge. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 403–410. Morgan Kaufmann Publishers Inc., 1995.

T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

J. Mooij and T. Claassen. Constraint-Based Causal Discovery using Partial Ancestral Graphs in the presence of Cycles. In *Proceedings of the Thirty-sixth Conference on Uncertainty in Artificial Intelligence*. PMLR, 2020.

P. Nandy, M. H. Maathuis, and T. S. Richardson. Estimating the effect of joint interventions from observational data in sparse high-dimensional settings. *The Annals of Statistics*, 45(2):647–674, 04 2017.
J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

J. Pearl. *Causality*. Cambridge University Press, 2009.

J. Pearl, D. Geiger, and T. Verma. Conditional independence and its representations. *Kybernetika*, 25(7):33–44, 1989.

E. Perković. Identifying causal effects in maximally oriented partially directed acyclic graphs. In *Proceedings of the Thirty-sixth Conference on Uncertainty in Artificial Intelligence*. PMLR, 2020.

E. Perković, M. Kalisch, and M. H. Maathuis. Interpreting and using CPDAGs with background knowledge. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*. AUAI press, 2017.

J. Peters and P. Bühlmann. Identifiability of Gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228, 2013.

J. Peters, J. M. Mooij, D. Janzing, and B. Schölkopf. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15(1):2009–2053, 2014.

T. Richardson and P. Spirtes. Ancestral graph Markov models. *The Annals of Statistics*, 30(4):962–1030, 08 2002.

D. J. Rose and R. E. Tarjan. Algorithmic aspects of vertex elimination on directed graphs. Technical report, Stanford, CA, USA, 1975.

A. Roumpelaki, G. Borboudakis, S. Triantafiellou, and I. Tsamardinos. Marginal causal consistency in constraint-based causal learning. In *Proceedings of the UAI 2016 Workshop on Causation: Foundation to Application*, Foundation to Application,number 1792 in CEUR Workshop Proceedings, page 39–47, 2016.

S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. Kerminen. A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(Oct):2003–2030, 2006.

S. Shimizu, T. Inazumi, Y. Sogawa, A. Hyvärinen, Y. Kawahara, T. Washio, P. O. Hoyer, and K. Bollen. Directlingam: A direct method for learning a linear non-Gaussian structural equation model. *Journal of Machine Learning Research*, 12(Apr):1225–1248, 2011.
A. Singh and A. Moore. Finding optimal Bayesian networks by dynamic programming. Technical report, 2005.

P. Spirtes and C. Glymour. An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review*, 9(1):62–72, 1991.

I. Tsamardinos and C. F. Aliferis. Towards principled feature selection: Relevancy, filters and wrappers. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*. Morgan Kaufmann Publishers, 2003.

I. Tsamardinos, C. F. Aliferis, and A. R. Statnikov. Algorithms for large scale Markov blanket discovery. In *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, pages 376–381. AAAI Press, 2003.

C. Wang, Y. Zhou, Q. Zhao, and Z. Geng. Discovering and orienting the edges connected to a target variable in a DAG via a sequential local learning approach. *Computational Statistics & Data Analysis*, 77:252 – 266, 2014.

J. Witte, L. Henckel, M. H. Maathuis, and V. Didelez. On efficient adjustment in causal graphs. *Journal of Machine Learning Research*, 21(246):1–45, 2020.

Y. Wu, L. Zhang, and X. Wu. Counterfactual fairness: Unidentification, bound and algorithm. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1438–1444. International Joint Conferences on Artificial Intelligence Organization, 2019.

J. Xiang and S. Kim. A* lasso for learning a sparse Bayesian network structure for continuous variables. In *Advances in Neural Information Processing Systems 26*, pages 2418–2426. Curran Associates, Inc., 2013.

C. Yuan, B. Malone, and X. Wu. Learning optimal Bayesian networks using A* search. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI-11*, pages 2186–2191. International Joint Conferences on Artificial Intelligence Organization, 2011.

J. Zhang. *Causal Inference and Reasoning in Causally Insufficient Systems*. PhD thesis, Carnegie Mellon University, 2006.
J. Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16):1873 – 1896, 2008.

K. Zhang and A. Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI press, 2009.

K. Zhang, M. Gong, P. Stojanov, B. Huang, Q. Liu, and C. Glymour. Domain adaptation as a problem of inference on graphical models. In *Advances in Neural Information Processing Systems*, volume 33, pages 4965–4976. Curran Associates, Inc., 2020.

X. Zheng, B. Aragam, P. K. Ravikumar, and E. P. Xing. DAGs with no tears: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems 31*, pages 9472–9483. Curran Associates, Inc., 2018.