
Causal Identification under Markov Equivalence

Amin Jaber

Computer Science Department
Purdue University, IN, USA
jaber0@purdue.edu

Jiji Zhang

Philosophy Department
Lingnan University, NT, HK
jjizhang@ln.edu.hk

Elias Bareinboim

Computer Science Department
Purdue University, IN, USA
eb@purdue.edu

Abstract

Assessing the magnitude of cause-and-effect relations is one of the central challenges found throughout the empirical sciences. The problem of identification of causal effects is concerned with determining whether a causal effect can be computed from a combination of observational data and substantive knowledge about the domain under investigation, which is formally expressed in the form of a causal graph. In many practical settings, however, the knowledge available for the researcher is not strong enough so as to specify a unique causal graph. Another line of investigation attempts to use observational data to learn a qualitative description of the domain called a Markov equivalence class, which is the collection of causal graphs that share the same set of observed features. In this paper, we marry both approaches and study the problem of causal identification from an equivalence class, represented by a partial ancestral graph (PAG). We start by deriving a set of graphical properties of PAGs that are carried over to its induced subgraphs. We then develop an algorithm to compute the effect of an arbitrary set of variables on an arbitrary outcome set. We show that the algorithm is strictly more powerful than the current state of the art found in the literature.

1 INTRODUCTION

Science is about explaining the mechanisms underlying a phenomenon that is being investigated. One of the marks imprinted by these mechanisms in reality is cause and effect relationships. Systematically discovering the existence, and magnitude, of causal relations constitutes,

therefore, a central task in scientific domains. The value of inferring causal relationships is also tremendous in other, more practical domains, including, for example, engineering and business, where it is often crucial to understand how to bring about a specific change when a constrained amount of controllability is in place. If our goal is to build AI systems that can act and learn autonomously, formalizing the principles behind causal inference, so that these systems can leverage them, is a fundamental requirement (Pearl and Mackenzie, 2018).

One prominent approach to infer causal relations leverages a combination of substantive knowledge about the domain under investigation, usually encoded in the form of a causal graph, with observational (non-experimental) data (Pearl, 2000; Spirtes et al., 2001; Bareinboim and Pearl, 2016). A sample causal graph is shown in Fig. 1a such that the nodes represent variables, directed edges represent direct causal relation from tails to heads, and bi-directed arcs represent the presence of unobserved (latent) variables that generate a spurious association between the variables, also known as *confounding bias* (Pearl, 1993). The task of determining whether an interventional (experimental) distribution can be computed from a combination of observational and experimental data together with the causal graph is known as the problem of identification of causal effects (identification, for short). For instance, a possible task in this case is to identify the effect of $do(X=x)$ on $V_4=v_4$, i.e. $P_x(v_4)$, given the causal graph in Fig. 1a and data from the observational distribution $P(x, v_1, \dots, v_4)$.

The problem of identification has been extensively studied in the literature, and a number of criteria have been established (Pearl, 1993; Galles and Pearl, 1995; Kuroki and Miyakawa, 1999; Tian and Pearl, 2002; Huang and Valtorta, 2006; Shpitser and Pearl, 2006; Bareinboim and Pearl, 2012), which include the celebrated back-door criterion and the do-calculus (Pearl, 1995). Despite their power, these techniques require a fully specified causal graph, which is not always available in practical settings.

Another line of investigation attempts to learn a qualitative description of the system, which in the ideal case would lead to the “true” data-generating model, the blueprint underlying the phenomenon being investigated. These efforts could certainly be deemed more “data-driven” and aligned with the zeitgeist in machine learning. In practice, however, it is common that only an equivalence class of causal models can be consistently inferred from observational data (Verma, 1993; Spirtes et al., 2001; Zhang, 2008b). One useful characterization of such an equivalence class comes under the rubric of *partial ancestral graphs (PAGs)*, which will be critical to our work. Fig. 1 shows the PAG (right) that can be inferred from observational data that is consistent with the true causal model (left). The directed edges in a PAG signify ancestral relations (not necessarily direct) and circle marks stand for structural uncertainty.

In this paper, we analyze the marriage of these two lines of investigation, where the structural invariance learned in the equivalence class will be used as input to identify the strength of causal effect relationships, if possible. Identification from an equivalence class is considerably more challenging than from a single diagram due to the structural uncertainty regarding both the direct causal relations among the variables and the presence of latent variables that confounds causal relations between observed variables. Still, there is a growing interest in identifiability results in this setting (Maathuis et al., 2010). Zhang (2007) extended the do-calculus to PAGs. In practice, however, it is in general computationally hard to decide whether there exists (and, if so, find) a sequence of applications of the rules of the generalized calculus to identify the interventional distribution. Perković et al. (2015) generalized the back-door criterion to PAGs, and provided a sound and complete algorithm to find a back-door admissible set, should such a set exist. However, in practice, the back-door criterion is not as powerful as the do-calculus, since no adjustment set exists for many identifiable causal effects. Jaber et al. (2018b) generalized the work of (Tian and Pearl, 2002) and devised a graphical criterion to identify causal effects with singleton interventions in PAGs.¹

Building on this work, we develop here a decomposition strategy akin to the one introduced in (Tian, 2002) to identify causal effects given a PAG. Our proposed approach is computationally more attractive than the do-calculus as it provides a systematic procedure to identify

¹Another possible approach is based on SAT (boolean constraint satisfaction) solvers (Hyttinen et al., 2015). Given its somewhat distinct nature, a closer comparison lies outside the scope of this paper. We note, however, that an open research direction would be to translate our systematic approach into logical rules so as to help improving the solver’s scalability.

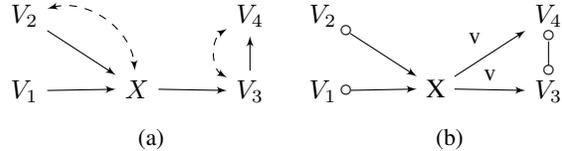


Figure 1: A causal model (left) and the inferred PAG (right).

a causal effect, if identifiable. It is also more powerful than the generalized adjustment criterion, as we show later. More specifically, our main contributions are:

1. We study some critical properties of PAGs and show that they also hold in induced subgraphs of a PAG over an arbitrary subset of nodes. We further study Tian’s *c*-component decomposition and relax it to PAGs (when only partial knowledge about the ancestral relations and *c*-components is available).
2. We formulate a systematic procedure to compute the effect of an arbitrary set of intervention variables on an arbitrary outcome set from a PAG and observational data. We show that this algorithm is strictly more powerful than the adjustment criterion.

2 PRELIMINARIES

In this section, we introduce the basic notation and machinery used throughout the paper. Bold capital letters denote sets of variables, while bold lowercase letters stand for particular assignments to those variables.

Structural Causal Models. We use the language of Structural Causal Models (SCM) (Pearl, 2000, pp. 204–207) as our basic semantic framework. Formally, an SCM M is a 4-tuple $\langle U, V, F, P(u) \rangle$, where U is a set of exogenous (latent) variables and V is a set of endogenous (measured) variables. F represents a collection of functions $F = \{f_i\}$ such that each endogenous variable $V_i \in V$ is determined by a function $f_i \in F$, where f_i is a mapping from the respective domain of $U_i \cup Pa_i$ to V_i , $U_i \subseteq U$, $Pa_i \subseteq V \setminus V_i$. The uncertainty is encoded through a probability distribution over the exogenous variables, $P(u)$. A causal diagram associated with an SCM encodes the structural relations among $V \cup U$, in which an arrow is drawn from each member of $U_i \cup Pa_i$ to V_i . We constraint our results to recursive systems, which means that the corresponding diagram will be acyclic. The marginal distribution over the endogenous variables $P(v)$ is called observational, and factorizes according to the causal diagram, i.e.:

$$P(v) = \sum_u \prod_i P(v_i | pa_i, u_i) P(u)$$

Within the structural semantics, performing an action $X=x$ is represented through the do-operator, $do(X=x)$, which encodes the operation of replacing the original equation for X by the constant x and induces a submodel M_x . The resulting distribution is denoted by P_x , which is the main target for identification in this paper. For details on structural models, we refer readers to (Pearl, 2000).

Ancestral Graphs. We now introduce a graphical representation of equivalence classes of causal diagrams. A *mixed* graph can contain directed (\rightarrow) and bi-directed edges (\leftrightarrow). A is a *spouse* of B if $A \leftrightarrow B$ is present. An *almost directed cycle* happens when A is both a spouse and an ancestor of B . An *inducing path relative to \mathbf{L}* is a path on which every node $V \notin \mathbf{L}$ (except for the endpoints) is a collider on the path (i.e., both edges incident to V are into V) and every collider is an ancestor of an endpoint of the path. A mixed graph is *ancestral* if it doesn't contain a directed or almost directed cycle. It is *maximal* if there is no inducing path (relative to the empty set) between any two non-adjacent nodes. A *Maximal Ancestral Graph* (MAG) is a graph that is both ancestral and maximal. MAG models are closed under marginalization (Richardson and Spirtes, 2002).

In general, a causal MAG represents a set of causal models with the same set of observed variables that entail the same independence and ancestral relations among the observed variables. Different MAGs may be Markov equivalent in that they entail the exact same independence model. A partial ancestral graph (PAG) represents an equivalence class of MAGs $[\mathcal{M}]$, which shares the same adjacencies as every MAG in $[\mathcal{M}]$ and displays all and only the invariant edge marks.

Definition 1 (PAG). Let $[\mathcal{M}]$ be the Markov equivalence class of an arbitrary MAG \mathcal{M} . The PAG for $[\mathcal{M}]$, \mathcal{P} , is a partial mixed graph such that:

- i. \mathcal{P} has the same adjacencies as \mathcal{M} (and hence any member of $[\mathcal{M}]$) does.
- ii. An arrowhead is in \mathcal{P} iff shared by all MAGs in $[\mathcal{M}]$.
- iii. A tail is in \mathcal{P} iff shared by all MAGs in $[\mathcal{M}]$.
- iv. A mark that is neither an arrowhead nor a tail is recorded as a circle.

A PAG is learnable from the conditional independence and dependence relations among the observed variables and the FCI algorithm is a standard method to learn such an object (Zhang, 2008b). In short, a PAG represents an equivalence class of causal models with the same observed variables and independence model.

Graphical Notions. Given a DAG, MAG, or PAG, a path between X and Y is *potentially directed (causal)*

from X to Y if there is no arrowhead on the path pointing towards X . Y is called a *possible descendant* of X and X a *possible ancestor* of Y if there is a potentially directed path from X to Y . A set \mathbf{A} is (*descendant*) *ancestral* if no node outside \mathbf{A} is a possible (descendant) ancestor of any node in \mathbf{A} . Y is called a *possible child* of X , i.e. $Y \in \text{Ch}(X)$, and X a *possible parent* of Y , i.e. $X \in \text{Pa}(Y)$, if they are adjacent and the edge is not into X . For a set of nodes \mathbf{X} , we have $\text{Pa}(\mathbf{X}) = \cup_{X \in \mathbf{X}} \text{Pa}(X)$ and $\text{Ch}(\mathbf{X}) = \cup_{X \in \mathbf{X}} \text{Ch}(X)$. Given two sets of nodes \mathbf{X} and \mathbf{Y} , a path between them is called *proper* if one of the endpoints is in \mathbf{X} and the other is in \mathbf{Y} , and no other node on the path is in \mathbf{X} or \mathbf{Y} . For convenience, we use an asterisk (*) to denote any possible mark of a PAG ($\circ, >, -$) or a MAG ($>, -$). If the edge marks on a path between X and Y are all circles, we call the path a *circle path*.

A directed edge $X \rightarrow Y$ in a MAG or PAG is *visible* if there exists no DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$ in the corresponding equivalence class where there is an inducing path between X and Y that is into X relative to \mathbf{L} . This implies that a visible edge is not confounded ($X \leftarrow U_i \rightarrow Y$ doesn't exist). Which directed edges are visible is easily decidable by a graphical condition (Zhang, 2008a), so we simply mark visible edges by v . For brevity, we refer to any edge that is not a visible directed edge as *invisible*.

Identification Given a Causal DAG. Tian and Pearl (2002) presented an identification algorithm based on a decomposition strategy of the DAG into a set of so-called *c-components* (confounded components).

Definition 2 (C-Component). In a causal DAG, two observed variables are said to be in the same *c-component* if and only if they are connected by a bi-directed path, i.e. a path composed solely of such bi-directed treks as $V_i \leftarrow U_{ij} \rightarrow V_j$, where U_{ij} is an exogenous variable.

For convenience, we often refer to a bi-directed trek like $V_i \leftarrow U_{ij} \rightarrow V_j$ as a bi-directed edge between V_i and V_j (and U_{ij} is often left implicit). For any set $\mathbf{C} \subseteq \mathbf{V}$, we define the quantity $Q[\mathbf{C}]$ to denote the post-intervention distribution of \mathbf{C} under an intervention on $\mathbf{V} \setminus \mathbf{C}$:

$$Q[\mathbf{C}] = P_{\mathbf{v} \setminus \mathbf{c}}(\mathbf{c}) = \sum_{\mathbf{u}} \prod_{\{i | V_i \in \mathbf{C}\}} P(v_i | pa_i, u_i) P(u)$$

The significance of *c-components* and their decomposition is evident from (Tian, 2002, Lemmas 10, 11), which are the basis of Tian's identification algorithm.

3 REVISIT IDENTIFICATION IN DAGS

We revisit the identification results in DAGs, focusing on Tian's algorithm (Tian, 2002). Our goal here is to have an amenable algorithm that allows the incorporation of the structural uncertainties arising in the equivalence class.

Let \mathcal{D}_A denote the (induced) subgraph of a DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$ over $\mathbf{A} \subseteq \mathbf{V}$ and the latent parents of \mathbf{A} (i.e. $\text{Pa}(\mathbf{A}) \cap \mathbf{L}$). The original algorithm (Alg. 5 in (Tian, 2002)) alternately applies Lemmas 10 and 11 in (Tian, 2002) until a solution is derived or a failure condition is triggered. We rewrite this algorithm with a more local, atomic criterion based on the following results.

Definition 3 (Composite C-Component). *Given a DAG that decomposes into c-components S_1, \dots, S_k , $k \geq 1$, a composite c-component is the union of one or more of these c-components.*

Lemma 1. *Given a DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$, $\mathbf{X} \subset \mathbf{T} \subseteq \mathbf{V}$, and $P_{\mathbf{V} \setminus \mathbf{T}}$ the interventional distribution of $\mathbf{V} \setminus \mathbf{T}$ on \mathbf{T} . Let $S^{\mathbf{X}}$ denote a composite c-component containing \mathbf{X} in $\mathcal{D}_{\mathbf{T}}$. If \mathbf{X} is a descendant set in $\mathcal{D}_{S^{\mathbf{X}}}$, then $Q[\mathbf{T} \setminus \mathbf{X}]$ is identifiable and given by*

$$Q[\mathbf{T} \setminus \mathbf{X}] = \frac{P_{\mathbf{V} \setminus \mathbf{T}}}{Q[S^{\mathbf{X}}]} \times \sum_{\mathbf{x}} Q[S^{\mathbf{X}}] \quad (1)$$

Proof. By (Tian, 2002, Lemma 11), $Q[\mathbf{T}]$ decomposes as follows.

$$Q[\mathbf{T}] = Q[\mathbf{T} \setminus S^{\mathbf{X}}] \times Q[S^{\mathbf{X}}] = \frac{Q[\mathbf{T}]}{Q[S^{\mathbf{X}}]} \times Q[S^{\mathbf{X}}]$$

$Q[S^{\mathbf{X}}]$ is computable from $P_{\mathbf{V} \setminus \mathbf{T}}$ using Lemma 11 in (Tian, 2002), and $Q[S^{\mathbf{X}} \setminus \mathbf{X}]$ is computable from $Q[S^{\mathbf{X}}]$ using (Tian, 2002, Lemma 10) as \mathbf{X} is a descendant set in $\mathcal{D}_{S^{\mathbf{X}}}$. Therefore,

$$Q[\mathbf{T} \setminus \mathbf{X}] = \frac{P_{\mathbf{V} \setminus \mathbf{T}}}{Q[S^{\mathbf{X}}]} \cdot Q[S^{\mathbf{X}} \setminus \mathbf{X}] = \frac{P_{\mathbf{V} \setminus \mathbf{T}}}{Q[S^{\mathbf{X}}]} \cdot \sum_{\mathbf{x}} Q[S^{\mathbf{X}}] \quad \square$$

The next result follows directly when \mathbf{X} is a singleton.

Corollary 1. *Given a DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$, $X \in \mathbf{T} \subseteq \mathbf{V}$, and $P_{\mathbf{V} \setminus \mathbf{T}}$. If X is not in the same c-component with a child in $\mathcal{D}_{\mathbf{T}}$, then $Q[\mathbf{T} \setminus \{X\}]$ is identifiable and given by*

$$Q[\mathbf{T} \setminus \{X\}] = \frac{P_{\mathbf{V} \setminus \mathbf{T}}}{Q[S^X]} \times \sum_x Q[S^X] \quad (2)$$

where S^X is the c-component of X in $\mathcal{D}_{\mathbf{T}}$.

The significance of Corol. 1 stems from the fact that it can be used to rewrite the identification algorithm in a step-wise fashion, which is shown in Algorithm 1. The same is equivalent to the original algorithm since neither one of Lemmas 10 nor 11 in (Tian, 2002) is applicable whenever Corol. 1 is not applicable, which is shown by Lemmas 2 and 3. This result may not be surprising since Corol. 1 follows from the application of these lemmas.

Algorithm 1: ID(\mathbf{x}, \mathbf{y}) given DAG \mathcal{G}

input : two disjoint sets $\mathbf{X}, \mathbf{Y} \subset \mathbf{V}$

output: Expression for $P_{\mathbf{x}}(\mathbf{y})$ or FAIL

1. Let $\mathbf{D} = \text{An}(\mathbf{Y})_{\mathcal{G}_{\mathbf{V} \setminus \mathbf{X}}}$
2. Let the c-components of $\mathcal{G}_{\mathbf{D}}$ be $\mathbf{D}_i, i = 1, \dots, k$
3. $P_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{d} \setminus \mathbf{y}} \prod_i \text{Identify}(\mathbf{D}_i, \mathbf{V}, P)$

Function Identify ($\mathbf{C}, \mathbf{T}, Q = Q[\mathbf{T}]$) :

```

if  $\mathbf{C} = \mathbf{T}$  then
  | return  $Q[\mathbf{T}]$ ;
end
/* Let  $S^B$  be the c-component of  $\{B\}$  in  $\mathcal{G}_{\mathbf{T}}$  */
if  $\exists B \in \mathbf{T} \setminus \mathbf{C}$  such that  $S^B \cap \text{Ch}(B) = \emptyset$  then
  | Compute  $Q[\mathbf{T} \setminus \{B\}]$  from  $Q$ ; // Corollary 1
  | return Identify( $\mathbf{C}, \mathbf{T} \setminus \{B\}, Q[\mathbf{T} \setminus \{B\}]$ );
else
  | throw FAIL;
end

```

Lemma 2. *Given a DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$, $\mathbf{C} \subset \mathbf{T} \subseteq \mathbf{V}$. If $\mathbf{A} = \text{An}(\mathbf{C})_{\mathcal{D}_{\mathbf{T}}} \neq \mathbf{T}$, then there exist some node $X \in \mathbf{T} \setminus \mathbf{A}$ such that X is not in the same c-component with any child in $\mathcal{D}_{\mathbf{T}}$.*

Proof. If $\mathbf{A} \neq \mathbf{T}$, then $\mathbf{T} \setminus \mathbf{A}$ is a non-empty set where none of the nodes is an ancestor of \mathbf{A} . Since the graph is acyclic, then at least one node of $\mathbf{T} \setminus \mathbf{A}$ is with no children. Hence, the above conclusion follows. \square

Lemma 3. *Given a DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$, $\mathbf{C} \subset \mathbf{T} \subseteq \mathbf{V}$, and assume $\mathcal{D}_{\mathbf{C}}$ is a single c-component. If $\mathcal{D}_{\mathbf{T}}$ partitions into c-components $\mathbf{S}_1 \dots \mathbf{S}_k$, where $k > 1$, then there exists some node $X \in \mathbf{S}_i$ such that $\mathbf{C} \not\subseteq \mathbf{S}_i$ and X is not in the same c-component with any child in $\mathcal{D}_{\mathbf{T}}$.*

Proof. Subgraph $\mathcal{D}_{\mathbf{S}_i}$ is acyclic, so there must exist some node (X) that doesn't have any children in $\mathcal{D}_{\mathbf{S}_i}$. Since \mathbf{S}_i is one of the c-components in $\mathcal{D}_{\mathbf{T}}$, then X is not in the same c-component with any of its children in $\mathcal{D}_{\mathbf{T}}$. \square

The revised algorithm requires checking an atomic criterion at every instance of the recursive routine Identify. This might not be crucial when the precise causal diagram is known and induced subgraphs preserve a complete graphical characterization of the c-components and the ancestral relations between the nodes. The latter, unfortunately, doesn't hold when the model is an equivalence class represented by a PAG.²

²We thank a reviewer for bringing to our attention a similar formulation of Alg. 1 (Richardson et al., 2017, Thm. 60).

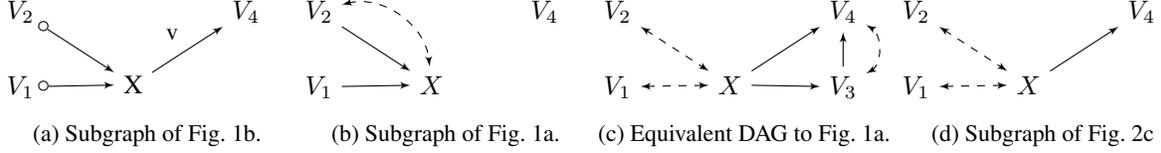


Figure 2: Example for properties discussed in Section 4

4 PAG-SUBGRAPH PROPERTIES

Evidently, induced subgraphs of the original causal model play a critical role in identification (cf Alg. 1). It is natural to expect that in the generalized setting we study here, induced subgraphs of the given PAG will also play an important role. An immediate challenge, however, is that a subgraph of a PAG \mathcal{P} over \mathbf{V} induced by $\mathbf{A} \subseteq \mathbf{V}$ is, in general, not a PAG that represents a full Markov equivalence class. In particular, if $\mathcal{D}(\mathbf{V}, \mathbf{L})$ is a DAG in the equivalence class represented by \mathcal{P} , $\mathcal{P}_{\mathbf{A}}$ is in general not the PAG that represents the equivalence class of $\mathcal{D}_{\mathbf{A}}$. To witness, let \mathcal{D} and \mathcal{P} denote the DAG and the corresponding PAG in Figure 1, respectively, and let $\mathbf{A} = \{V_1, V_2, X, V_4\}$. The induced subgraph of \mathcal{P} over \mathbf{A} (Fig. 2a) does not represent the equivalence class of the corresponding induced subgraph of \mathcal{D} (Fig. 2b). Despite this subtlety, we establish a few facts below showing that for any $\mathbf{A} \subseteq \mathbf{V}$ and any DAG \mathcal{D} in the equivalence class represented by \mathcal{P} , some information about $\mathcal{D}_{\mathbf{A}}$, which is particularly relevant to identification, can be read off from $\mathcal{P}_{\mathbf{A}}$.

Proposition 1. *Let \mathcal{P} be a PAG over \mathbf{V} , and $\mathcal{D}(\mathbf{V}, \mathbf{L})$ be any DAG in the equivalence class represented by \mathcal{P} . Let $X \neq Y$ be two nodes in $\mathbf{A} \subseteq \mathbf{V}$. If X is an ancestor of Y in $\mathcal{D}_{\mathbf{A}}$, then X is a possible ancestor of Y in $\mathcal{P}_{\mathbf{A}}$.*

Proof. If X is an ancestor of Y in $\mathcal{D}_{\mathbf{A}}$, then there is a path p in $\mathcal{D}_{\mathbf{A}}$ composed of nodes $\langle X = V_0, \dots, Y = V_m \rangle$, $m \geq 1$ such that $V_i \in \mathbf{A}$ and $V_i \rightarrow V_{i+1}$, $0 \leq i < m$. Path p is obviously also present in \mathcal{D} , and consequently the corresponding MAG \mathcal{M} . Hence, p corresponds to a possibly directed path in \mathcal{P} . Since all the nodes along p are in \mathbf{A} , then p is present in $\mathcal{P}_{\mathbf{A}}$ and so X is a possible ancestor of Y in $\mathcal{P}_{\mathbf{A}}$. \square

This simple proposition guarantees that possible-ancestral relationship in $\mathcal{P}_{\mathbf{A}}$ subsumes ancestral relationship in $\mathcal{D}_{\mathbf{A}}$ for every \mathcal{D} in the class represented by \mathcal{P} . This is illustrated by $\mathcal{D}_{\mathbf{A}}$ and $\mathcal{P}_{\mathbf{A}}$ in Figures 2a and 2b.

Given an induced subgraph of a PAG, $\mathcal{P}_{\mathbf{A}}$, a directed edge $X \rightarrow Y$ in $\mathcal{P}_{\mathbf{A}}$ is said to be *visible* if for every DAG \mathcal{D} in the class represented by \mathcal{P} , there is no inducing path in $\mathcal{D}_{\mathbf{A}}$ between X and Y relative to the latent nodes in $\mathcal{D}_{\mathbf{A}}$ that is into X .

Lemma 4. *Let \mathcal{P} be a PAG over \mathbf{V} , and $\mathcal{P}_{\mathbf{A}}$ be an induced subgraph of \mathcal{P} over $\mathbf{A} \subseteq \mathbf{V}$. For every $X \rightarrow Y$ in $\mathcal{P}_{\mathbf{A}}$, if it is visible in \mathcal{P} , then it remains visible in $\mathcal{P}_{\mathbf{A}}$.*

Proof. Let $\mathcal{D}(\mathbf{V}, \mathbf{L})$ be any causal model in the equivalence class represented by \mathcal{P} , and let $X \rightarrow Y$ be a visible edge in \mathcal{P} , $X, Y \in \mathbf{A}$. Then, there is no inducing path between X and Y relative to \mathbf{L} that is into X in \mathcal{D} . It follows that no such inducing path (relative to the latent nodes in $\mathcal{D}_{\mathbf{A}}$) exists in the subgraph $\mathcal{D}_{\mathbf{A}}$. \square

Visibility is relevant for identification because it implies absence of confounding, which is the major obstacle to identification. Lemma 4 shows that an edge in an induced subgraph that is visible in the original PAG also implies absence of confounding in the induced subgraphs. Interestingly, note that a directed edge $X \rightarrow Y$ in $\mathcal{P}_{\mathbf{A}}$, visible or not, does not imply that X is an ancestor of Y in $\mathcal{D}_{\mathbf{A}}$ for every \mathcal{D} in the class represented by \mathcal{P} . For example, X is not an ancestor of V_4 in Fig. 2b, even though $X \rightarrow V_4$ is a visible edge in Fig. 2a.

Definition 4 (PC-Component). *In a MAG, a PAG, or any of its induced subgraphs, two nodes X and Y are in the same possible c-component (pc-component) if there is a path between the two nodes such that (1) all non-endpoint nodes along the path are colliders, and (2) none of the edges is visible.*

As alluded earlier, a c-component in a causal graph plays a central role in identification. The following proposition establishes a graphical condition in an induced subgraph $\mathcal{P}_{\mathbf{A}}$ that is necessary for two nodes being in the same c-component in $\mathcal{D}_{\mathbf{A}}$ for some DAG \mathcal{D} represented by \mathcal{P} .

Proposition 2. *Let \mathcal{P} be a PAG over \mathbf{V} , and $\mathcal{D}(\mathbf{V}, \mathbf{L})$ be any DAG in the equivalence class represented by \mathcal{P} . For any $X, Y \in \mathbf{A} \subseteq \mathbf{V}$, if X and Y are in the same c-component in $\mathcal{D}_{\mathbf{A}}$, then X and Y are in the same pc-component in $\mathcal{P}_{\mathbf{A}}$.*

Proof Sketch. If X and Y are in the same c-component in $\mathcal{D}_{\mathbf{A}}$, then there is a path p in $\mathcal{D}_{\mathbf{A}}$ composed of nodes $\langle X = V_0, \dots, Y = V_m \rangle$, $m \geq 1$, such that $V_i \in \mathbf{A}$ and $V_i \leftarrow L_{i,i+1} \rightarrow V_{i+1}$, $0 \leq i < m$. We prove that X and Y are in the same pc-component in \mathcal{M} , the MAG of \mathcal{D} over \mathbf{V} , due to a path p' over a subsequence of p . We then

show that X and Y are in the same pc-component in \mathcal{P} , the PAG of \mathcal{M} , due to a path p^* over a subsequence of p' . Since all the nodes along p^* are in \mathbf{A} , then p^* is present in $\mathcal{P}_{\mathbf{A}}$, and so X and Y are in the same pc-component in $\mathcal{P}_{\mathbf{A}}$. Due to space constraints, the complete proofs are provided in (Jaber et al., 2018a). \square

This result provides a sufficient condition for *not* belonging to the same c-component in any of the relevant causal graphs. In Fig. 2a, for example, V_1 and V_4 or X and V_4 are not in the same pc-component, which implies by Prop. 2 that they are not in the same c-component in $\mathcal{D}_{\mathbf{A}}$ for any DAG \mathcal{D} in the equivalence class represented by the PAG in Fig. 1b.

As a special case of Def. 4, we define the following notion, which will prove useful later on.

Definition 5 (DC-Component). *In a MAG, a PAG, or any of its induced subgraphs, two nodes X and Y are in the same definite c-component (dc-component) if they are connected with a bi-directed path, i.e. a path composed solely of bi-directed edges.*

One challenge with the notion of *pc-component* is that it is not transitive as *c-component* is. Consider the PAG $V_1 \circ\text{---}\circ V_2 \circ\text{---}\circ V_3$. Here, V_1 and V_2 are in the same pc-component, V_2 and V_3 are in the same pc-component, however, V_1 and V_3 are not in the same pc-component. Hence, we define a notion that is a transitive closure of the notion of pc-component, which will prove instrumental to our goal.

Definition 6 (CPC-Component). *Let \mathcal{P} denote a PAG or a corresponding induced subgraph. Nodes X and Y are in the same composite pc-component in \mathcal{P} , denoted *cpc-component*, if there exist a sequence of nodes $\langle X = V_0, \dots, Y = V_m \rangle$, $m \geq 1$, such that V_i and V_{i+1} are in the same pc-component, $0 \leq i < m$.*

It follows from the above definition that a PAG or an induced subgraph \mathcal{P} can be decomposed into unique sets of cpc-components. For instance, the cpc-components in Fig. 2a are $\mathbf{S}_1 = \{V_1, V_2, X\}$ and $\mathbf{S}_2 = \{V_4\}$. The significance of a *cpc-component* is that it corresponds to a composite c-component in the relevant causal graphs as shown in the following proposition.

Proposition 3. *Let \mathcal{P} be a PAG over \mathbf{V} , $\mathcal{D}(\mathbf{V}, \mathbf{L})$ be any DAG in the equivalence class represented by \mathcal{P} , and $\mathbf{A} \subseteq \mathbf{V}$. If $\mathbf{C} \subseteq \mathbf{A}$ is a cpc-component in $\mathcal{P}_{\mathbf{A}}$, then \mathbf{C} is a composite c-component in $\mathcal{D}_{\mathbf{A}}$.*

Proof. According to Definition 6, \mathbf{C} includes all the nodes that are in the same pc-component with some node in \mathbf{C} in $\mathcal{P}_{\mathbf{A}}$. It follows from the contrapositive of Prop. 2 that no node outside \mathbf{C} is in the same c-component with

Algorithm 2: PTO Algorithm

input : PAG \mathcal{P} over \mathbf{V}

output: PTO over \mathcal{P}

1- Create singleton buckets \mathbf{B}_i each containing $V_i \in \mathbf{V}$.

2- Merge buckets \mathbf{B}_i and \mathbf{B}_j if there is a circle edge between them ($\mathbf{B}_i \ni X \circ\text{---}\circ Y \in \mathbf{B}_j$).

3- **while** set of buckets (\mathbf{B}) is not empty **do**

 (i) Extract \mathbf{B}_i with only arrowheads incident on it.

 (ii) Remove edges between \mathbf{B}_i and other buckets.

end

4- The partial order is $\mathbf{B}_1 < \mathbf{B}_2 < \dots < \mathbf{B}_m$ in reverse order of the bucket extraction. Hence, \mathbf{B}_1 is the last bucket extracted and \mathbf{B}_m is the first bucket extracted.

any node in \mathbf{C} in $\mathcal{D}_{\mathbf{A}}$. Hence, set \mathbf{C} represents a composite c-component in $\mathcal{D}_{\mathbf{A}}$ by Definition 3. \square

Recall that the algorithm for identification given a DAG uses a topological order over the nodes. Similarly, the algorithm we design for PAGs will depend on some (partial) topological order. Thanks to the possible presence of circle edges ($\circ\text{---}\circ$) in a PAG, in general, there may be no complete topological order that is valid for all DAGs in the equivalence class. Algorithm 2 presents a procedure to derive a *partial topological order* over the nodes in a PAG, using buckets of nodes that are connected with circle paths (Jaber et al., 2018b). This algorithm remains valid over an induced subgraph of a PAG. To show this, the following lemma is crucial:

Lemma 5. *Let \mathcal{P} be a PAG over \mathbf{V} , and $\mathcal{P}_{\mathbf{A}}$ be the induced subgraph over $\mathbf{A} \subseteq \mathbf{V}$. For any three nodes A, B, C , if $A \ast \rightarrow B \circ\text{---}\ast C$, then there is an edge between A and C with an arrowhead at C , namely, $A \ast \rightarrow C$. Furthermore, if the edge between A and B is $A \rightarrow B$, then the edge between A and C is either $A \rightarrow C$ or $A \circ\text{---}\rightarrow C$ (i.e., it is not $A \leftrightarrow C$).*

Proof. Lemma 3.3.1 of (Zhang, 2006) establishes the above property for every PAG. By the definition of an induced subgraph, the property is preserved in $\mathcal{P}_{\mathbf{A}}$. \square

Thus, a characteristic feature of PAGs carries over to their induced subgraphs. It follows that Algorithm 2 is sound for induced subgraphs as well.

Proposition 4. *Let \mathcal{P} be a PAG over \mathbf{V} , and let $\mathcal{P}_{\mathbf{A}}$ be the subgraph of \mathcal{P} induced by $\mathbf{A} \subseteq \mathbf{V}$. Then, Algorithm 2 is sound over $\mathcal{P}_{\mathbf{A}}$, in the sense that the partial order is valid with respect to $\mathcal{D}_{\mathbf{A}}$, for every DAG \mathcal{D} in the equivalence class represented by \mathcal{P} .*

Proof. Let D be any DAG in the equivalence class represented by \mathcal{P} . By Prop. 1, the possible-ancestral relations in \mathcal{P}_A subsume those present in \mathcal{D}_A . Hence, a partial topological order that is valid with respect to \mathcal{P}_A is valid with respect to \mathcal{D}_A . The correctness of Alg. 2 with respect to a PAG in (Jaber et al., 2018b) depends only on the property in Lemma 5, a proof of which is given in the Supplementary Materials for completeness. Therefore, thanks to Lemma 5, the algorithm is also sound with respect to an induced subgraph \mathcal{P}_A . \square

For example, for \mathcal{P}_A in Fig. 2a, a partial topological order over the nodes is $V_1 < V_2 < X < V_4$, which is valid for all the relevant DAGs.

With these results about induced subgraphs of a PAG, we are ready to develop a recursive approach for identification given a PAG, to which we now turn.

5 IDENTIFICATION IN PAGS

We start by formally defining the notion of identification given a PAG, which generalizes the model-specific notion (Pearl, 2000, pp. 70).

Definition 7. *Given a PAG \mathcal{P} over \mathbf{V} and a query $P_{\mathbf{x}}(\mathbf{y})$ where $\mathbf{X}, \mathbf{Y} \subset \mathbf{V}$, $P_{\mathbf{x}}(\mathbf{y})$ is identifiable given \mathcal{P} if and only if $P_{\mathbf{x}}(\mathbf{y})$ is identifiable given every DAG $\mathcal{D}(\mathbf{V}, \mathbf{L})$ in the Markov equivalence class represented by \mathcal{P} , and with the same expression.*

We first derive an atomic identification criterion analogous to Corollary 1. As seen in the algorithm for constructing a partial order (Alg. 2), a bucket or circle component in a PAG is for our purpose analogous to a single node in a DAG. Therefore, the following criterion targets a bucket \mathbf{X} rather than a single node.

Theorem 1. *Given a PAG \mathcal{P} over \mathbf{V} , a partial topological order $\mathbf{B}_1 < \dots < \mathbf{B}_m$ with respect to \mathcal{P} , a bucket $\mathbf{X} = \mathbf{B}_j \subset \mathbf{T} \subseteq \mathbf{V}$, for some $1 \leq j \leq m$, and $P_{\mathbf{v} \setminus \mathbf{t}}$ (i.e. $Q[\mathbf{T}]$), $Q[\mathbf{T} \setminus \mathbf{X}]$ is identifiable if and only if there does not exist $X \in \mathbf{X}$ such that X has a possible child $C \notin \mathbf{X}$ that is in the same pc-component as X in \mathcal{P}_T . If identifiable, then the expression is given by*

$$Q[\mathbf{T} \setminus \mathbf{X}] = \frac{P_{\mathbf{v} \setminus \mathbf{t}}}{\prod_{\{i | \mathbf{B}_i \subseteq S^{\mathbf{X}}\}} P_{\mathbf{v} \setminus \mathbf{t}}(\mathbf{B}_i | \mathbf{B}^{(i-1)})} \times \sum_{\mathbf{x}} \prod_{\{i | \mathbf{B}_i \subseteq S^{\mathbf{X}}\}} P_{\mathbf{v} \setminus \mathbf{t}}(\mathbf{B}_i | \mathbf{B}^{(i-1)}), \quad (3)$$

where $S^{\mathbf{X}} = \bigcup_{X \in \mathbf{X}} S^X$, S^X being the dc-component of X in \mathcal{P}_T , and $\mathbf{B}^{(i-1)}$ denoting the set of nodes preceding bucket \mathbf{B}_i in the partial order.

Proof Sketch. (if) Let \mathcal{D} be any DAG in the equivalence class represented by \mathcal{P} , \mathcal{D}_T be the induced subgraph over \mathbf{T} , and S' be the smallest composite c-component containing \mathbf{X} in \mathcal{D}_T . We show that \mathbf{X} is a descendant set in $\mathcal{D}_{S'}$. Suppose otherwise for the sake of contradiction. Then, there is a node $C \in S' \setminus \mathbf{X}$ such that C is a child of X_i and is in the same c-component with X_j , where $X_i, X_j \in \mathbf{X}$ and possibly $i = j$. By Prop. 2, X_j is in the same pc-component with C in \mathcal{P}_T . Let T_i be the node closest to X_j along the collider path in \mathcal{P}_T between X_j and C consistent with Def. 4. If the edge between X_j and T_i in \mathcal{P}_T is not into X_j , then X_j is in the same pc-component with a possible child as the edge is not visible. This violates the criterion stated in the theorem. Otherwise, the edge is $X_j \leftrightarrow T_i$ and there exist a bi-directed edge between T_i and every node in \mathbf{X} (which follows from Lemma 5). Hence, X_i is in the same pc-component with a possible child C in \mathcal{P}_T (Prop. 1), and the criterion stated in the theorem is violated again. Therefore, \mathbf{X} is a descendant set in $\mathcal{D}_{S'}$ and $Q[\mathbf{T} \setminus \mathbf{X}]$ is identifiable from $Q[\mathbf{T}]$ by Lemma 1. It remains to show that Eq. 3 is equivalent to Eq. 1 for \mathcal{D} . The details for this step are left to the Supplementary Material.

(only if) Suppose the criterion in question is not satisfied. Then some $X_i \in \mathbf{X}$ is in the pc-component with a possible child $C \notin \mathbf{X}$ in \mathcal{P}_T . The edge between X_i and C is $X_i \ast \rightarrow C$ as C is outside of \mathbf{X} . If the edge is not visible in \mathcal{P}_T , then this edge is not visible in \mathcal{P} (Lemma 4). Hence, we can construct a DAG \mathcal{D} in the equivalence class of \mathcal{P} where C is a child of X_i and the two nodes share a latent variable. The pair of sets $\mathbf{F} = \{X_i, C\}$ and $\mathbf{F}' = \{C\}$ form a so-called hedge for $Q[\mathbf{T} \setminus \mathbf{X}]$ and the effect is not identifiable in \mathcal{D} (Shpitser and Pearl, 2006, Theorem 4), and hence not identifiable given \mathcal{P} .

Otherwise, $X_i \rightarrow C$ is visible in \mathcal{P}_T . So, there is a collider path between X_i and C consistent with Def. 4 such that the two nodes are in the same pc-component. Let $p = \langle X_i = T_0, T_1, \dots, T_m = C \rangle$ denote the shortest such path in \mathcal{P}_T . If the edge between X_i and T_1 is not into X_i , then T_1 is a child of X_i and the proof follows as in the previous case. Otherwise, we have $X_i \leftrightarrow T_1$ and we can show that X_i is the only node along p that belongs to \mathbf{X} (details in the Supplementary Material). In \mathcal{P} , path p is present with $X_i \rightarrow C$ visible. Hence, we can construct a DAG \mathcal{D} in the equivalence class of \mathcal{P} such that C is a child of X_i and both are in the same c-component through a sequence of bi-directed edges along the corresponding nodes of p . The pair of sets $\mathbf{F} = \{X_i, T_1, \dots, T_m = C\}$ and $\mathbf{F}' = \{T_1, \dots, T_m = C\}$ form a hedge for $Q[\mathbf{T} \setminus \mathbf{X}]$ and the effect is not identifiable in \mathcal{D} , and hence it is not identifiable given \mathcal{P} . \square

Note that the above result simplifies into computing the

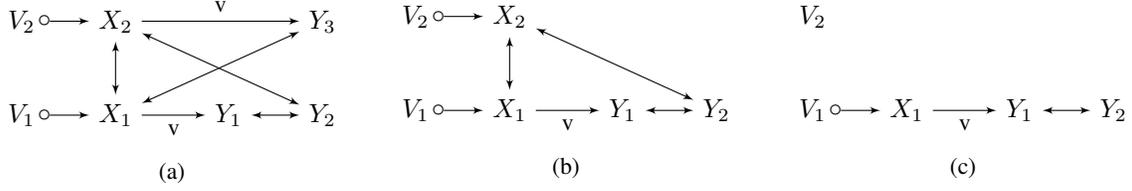


Figure 3: Sample PAG \mathcal{P} (left) and induced subgraphs used to identify $Q[\{Y_1, Y_2\}]$.

interventional distribution $P_{\mathbf{x}}$ whenever the input distribution is the observational distribution, i.e. $\mathbf{T} = \mathbf{V}$. Consider the query $P_{\mathbf{x}}(\mathbf{v} \setminus \{x\})$ over the PAG in Fig. 1b. The intervention node X is not in the same pc-component with any of its possible children (V_3 and V_4), hence the effect is identifiable and given by

$$\begin{aligned} P_{\mathbf{x}}(\mathbf{v} \setminus \{x\}) &= \frac{P(\mathbf{v})}{P(x|v_1, v_2)} \times \sum_{x'} P(x'|v_1, v_2) \\ &= P(v_1, v_2)P(v_4, v_5|v_1, v_2, x) \end{aligned}$$

Putting these observations together leads to the procedure we call **IDP**, which is shown in Alg. 3. In words, the main idea of **IDP** goes as follows. After receiving the sets \mathbf{X}, \mathbf{Y} , and a PAG \mathcal{P} , the algorithm starts the pre-processing steps: First, it computes \mathbf{D} , the set of possible ancestors of \mathbf{Y} in $\mathcal{P}_{\mathbf{V} \setminus \mathbf{X}}$. Second, it uses $\mathcal{P}_{\mathbf{D}}$ to partition set \mathbf{D} into cpc-components. Following the pre-processing stage, the procedure calls the subroutine `Identify` over each cpc-component \mathbf{D}_i to compute $Q[\mathbf{D}_i]$ from the observational distribution $P(\mathbf{V})$. The recursive routine basically checks for the presence of a bucket \mathbf{B} in $\mathcal{P}_{\mathbf{T}}$ that is a subset of the intervention nodes, i.e. $\mathbf{B} \subseteq \mathbf{T} \setminus \mathbf{C}$, and satisfies the conditions of Thm. 1. If found, it is able to successfully compute $Q[\mathbf{T} \setminus \mathbf{B}]$ using Eq. 3, and proceed with a recursive call. Alternatively, if such a bucket doesn't exist in $\mathcal{P}_{\mathbf{T}}$, then **IDP** throws a failure condition, since it's unable to identify the query. We show next that this procedure is, indeed, correct.

Theorem 2. *Algorithm IDP (Alg.3) is sound.*

Proof. Let $\mathcal{G}(\mathbf{V}, \mathbf{L})$ be any causal graph in the equivalence class of PAG \mathcal{P} over \mathbf{V} , and let $\mathbf{V}' = \mathbf{V} \setminus \mathbf{X}$. We have

$$P_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{v}' \setminus \mathbf{y}} P_{\mathbf{x}}(\mathbf{v}') = \sum_{\mathbf{v}' \setminus \mathbf{y}} Q[\mathbf{V}'] = \sum_{\mathbf{v}' \setminus \mathbf{d}} \sum_{\mathbf{d} \setminus \mathbf{y}} Q[\mathbf{V}']$$

By definition, \mathbf{D} is an ancestral set in $\mathcal{P}_{\mathbf{V}'}$, and hence it is ancestral in $\mathcal{G}_{\mathbf{V}'}$, by Prop. 1. So, we have the following by (Tian, 2002, Lemma 10):

$$P_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{d} \setminus \mathbf{y}} \sum_{\mathbf{v}' \setminus \mathbf{d}} Q[\mathbf{V}'] = \sum_{\mathbf{d} \setminus \mathbf{y}} Q[\mathbf{D}] \quad (4)$$

Algorithm 3: IDP(\mathbf{x}, \mathbf{y}) given PAG \mathcal{P}

input : two disjoint sets $\mathbf{X}, \mathbf{Y} \subset \mathbf{V}$

output: Expression for $P_{\mathbf{x}}(\mathbf{y})$ or FAIL

1. Let $\mathbf{D} = \text{An}(\mathbf{Y})_{\mathcal{P}_{\mathbf{V} \setminus \mathbf{X}}}$
2. Let the cpc-components of $\mathcal{P}_{\mathbf{D}}$ be $\mathbf{D}_i, i = 1, \dots, k$
3. $P_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{d} \setminus \mathbf{y}} \prod_i \text{Identify}(\mathbf{D}_i, \mathbf{V}, P)$

Function `Identify`($\mathbf{C}, \mathbf{T}, Q = Q[\mathbf{T}]$):

```

if  $\mathbf{C} = \mathbf{T}$  then
  | return  $Q[\mathbf{T}]$ ;
end
/* In  $\mathcal{P}_{\mathbf{T}}$ , let  $\mathbf{B}$  be a bucket, and  $\mathbf{C}^{\mathbf{B}}$  be the
pc-component of  $\mathbf{B}$  */
if  $\exists \mathbf{B} \subseteq \mathbf{T} \setminus \mathbf{C}$  such that  $\mathbf{C}^{\mathbf{B}} \cap \text{Ch}(\mathbf{B}) \subseteq \mathbf{B}$  then
  | Compute  $Q[\mathbf{T} \setminus \mathbf{B}]$  from  $Q$ ; // Theorem 1
  | return Identify( $\mathbf{C}, \mathbf{T} \setminus \mathbf{B}, Q[\mathbf{T} \setminus \mathbf{B}]$ );
else
  | throw FAIL;
end

```

Using Prop. 3, each cpc-component in $\mathcal{P}_{\mathbf{D}}$ corresponds to a composite c-component in $\mathcal{G}_{\mathbf{D}}$. Hence, Eq. 4 can be decomposed as follows by (Tian, 2002, Lemma 11).

$$P_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{d} \setminus \mathbf{y}} Q[\mathbf{D}] = \sum_{\mathbf{d} \setminus \mathbf{y}} \prod_i Q[\mathbf{D}_i] \quad (5)$$

Eq. 5 is equivalent to the decomposition we have in step 3 of Alg. 3, where we attempt to compute each $Q[\mathbf{D}_i]$ from P . Finally, the correctness of the recursive routine `Identify` follows from that of Theorem 1. \square

5.1 ILLUSTRATIVE EXAMPLE

Consider the query $P_{x_1, x_2}(y_1, y_2, y_3)$ given \mathcal{P} in Fig. 3a. We have $\mathbf{D} = \{Y_1, Y_2, Y_3\}$, and the cpc-components in $\mathcal{P}_{\mathbf{D}}$ are $\mathbf{D}_1 = \{Y_1, Y_2\}$ and $\mathbf{D}_2 = \{Y_3\}$. Hence, the problem reduces to computing $Q[\{Y_1, Y_2\}] \cdot Q[\{Y_3\}]$.

We start with the call `Identify`($\mathbf{D}_1, \mathbf{V}, P$). Consider the singleton bucket Y_3 the pc-component of which includes all the nodes in \mathcal{P} . This node satisfies the condition in `Identify` as it has no children, and we compute

$Q[\mathbf{V} \setminus \{Y_3\}]$ using Theorem 1.

$$\begin{aligned} Q[\mathbf{V} \setminus \{Y_3\}] &= \frac{P(\mathbf{v})}{P(y_1, y_2, y_3, x_1, x_2 | v_1, v_2)} \times \\ &\quad \sum_{y_3} P(y_1, y_2, y_3, x_1, x_2 | v_1, v_2) \\ &= P(v_1, v_2) \cdot P(y_1, y_2, x_1, x_2 | v_1, v_2) \\ &= P(y_1, y_2, x_1, x_2, v_1, v_2) \end{aligned} \quad (6)$$

In the next recursive call, $\mathbf{T}_1 = \mathbf{V} \setminus \{Y_3\}$, P_{y_3} corresponds to Eq. 6, and the induced subgraph $\mathcal{P}_{\mathbf{T}_1}$ is shown in Fig. 3b. Now, X_2 satisfies the criterion and we can compute $Q[\mathbf{T}_1 \setminus \{X_2\}]$ from $P_{y_3} = Q[\mathbf{T}_1]$, i.e.,

$$\begin{aligned} Q[\mathbf{T}_1 \setminus \{X_2\}] &= \frac{P_{y_3}}{P_{y_3}(y_1, y_2, x_1, x_2 | v_1, v_2)} \times \\ &\quad \sum_{x_2} P_{y_3}(y_1, y_2, x_1, x_2 | v_1, v_2) \\ &= P(y_1, y_2, x_1, v_1, v_2) \end{aligned} \quad (7)$$

Let $\mathbf{T}_2 = \mathbf{T}_1 \setminus \{X_2\}$, where the induced subgraph $\mathcal{P}_{\mathbf{T}_2}$ is shown in Fig. 3c. Now, X_1 satisfies the criterion and we can compute $Q[\mathbf{T}_2 \setminus \{X_1\}]$ from Eq. 7,

$$\begin{aligned} Q[\mathbf{T}_2 \setminus \{X_1\}] &= \frac{P_{y_3, x_2}}{P_{y_3, x_2}(x_1 | v_1, v_2)} \times \\ &\quad \sum_{x_1} P_{y_3, x_2}(x_1 | v_1, v_2) \\ &= \frac{P(v_1, v_2) \cdot P(y_1, y_2, x_1, v_1, v_2)}{P(x_1, v_1, v_2)} \\ &= P(v_1, v_2) \cdot P(y_1, y_2 | x_1, v_1, v_2) \end{aligned}$$

Choosing V_1 and V_2 in the next two recursive calls, we finally obtain the simplified expression:

$$Q[\{Y_1, Y_2\}] = P(y_1, y_2 | x_1)$$

Next, we solve for $Q[\mathbf{D}_2]$ and we get an expression analogous to that of $Q[\mathbf{D}_1]$. Hence, the final solution is:

$$P_{x_1, x_2}(y_1, y_2, y_3) = P(y_1, y_2 | x_1) \times P(y_3 | x_2)$$

5.2 COMPARISON TO STATE OF THE ART

In the previous section, we formulated an identification algorithm in PAGs for causal queries of the form $P_{\mathbf{x}}(\mathbf{y})$, $\mathbf{X}, \mathbf{Y} \subset \mathbf{V}$. A natural question arises about the expressiveness of the **IDP** in comparison with the state-of-the-art methods. One of the well established results in the literature is the adjustment method (Perković et al., 2015), which is complete whenever an adjustment set exists.

In the sequel, we formally show that the proposed algorithm subsumes the adjustment method.

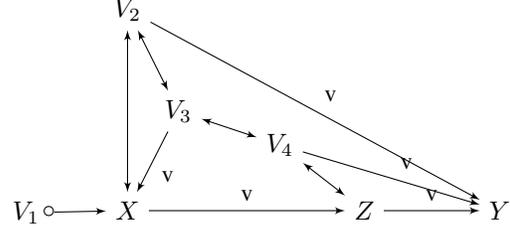


Figure 4: Query $P_{\mathbf{x}}(\mathbf{y})$ is identifiable by **IDP**.

Theorem 3. Let \mathcal{P} be a PAG over set \mathbf{V} and let $P_{\mathbf{x}}(\mathbf{y})$ be a causal query where $\mathbf{X}, \mathbf{Y} \subset \mathbf{V}$. If the distribution $P_{\mathbf{x}}(\mathbf{y})$ is not identifiable using **IDP** (Alg. 3), then the effect is not identifiable using the generalized adjustment criterion in (Perković et al., 2015).

Proof Sketch. Whenever **IDP** fails to identify some query, it is due to one of the recursive calls to `Identify`. We use the failing condition inside this call to systematically identify a proper definite status non-causal path from \mathbf{X} to \mathbf{Y} in \mathcal{P} that is m-connecting given set $\text{Adjust}(\mathbf{X}, \mathbf{Y}, \mathcal{P})$ (Perković et al., 2016, Def. 4.1). As this set fails to satisfy the adjustment criterion, then there exist no adjustment set relative to the pair (\mathbf{X}, \mathbf{Y}) in \mathcal{P} (Perković et al., 2016, Cor. 4.4). The details of the proof are left to the Supplementary Material. \square

Based on this result, one may wonder whether these algorithms are, after all, just equivalent. In reality, **IDP** captures strictly more identifiable effects than the adjustment criterion. To witness, consider the PAG in Fig. 4 and note that the causal distribution $P_{\mathbf{x}}(\mathbf{y})$ is identifiable by **IDP** but not by adjustment in this case.

6 CONCLUSION

We studied the problem of identification of interventional distributions in Markov equivalence classes represented by PAGs. We first investigated graphical properties for induced subgraphs of PAGs over an arbitrary subset of nodes with respect to induced subgraphs of DAGs that are in the equivalence class. We believe that these results can be useful to general tasks related to causal inference from equivalence classes. We further developed an identification algorithm in PAGs and proved it to subsume the state-of-the-art adjustment method.

Acknowledgments

We thank Sanghack Lee and the reviewers for all the feedback provided. Bareinboim and Jaber are supported in parts by grants from NSF IIS-1704352 and IIS-1750807 (CAREER). Zhang is supported in part by the Research Grants Council of Hong Kong under the General Research Fund LU13600715.

References

- Elias Bareinboim and Judea Pearl. Causal inference by surrogate experiments: z-identifiability. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, 2012.
- Elias Bareinboim and Judea Pearl. Causal inference and the data-fusion problem. *Proc. Natl. Acad. Sci.*, 113: 7345–7352, 2016.
- D. Galles and J. Pearl. Testing identifiability of causal effects. In P. Besnard and S. Hanks, editors, *Uncertainty in Artificial Intelligence 11*, pages 185–195. Morgan Kaufmann, San Francisco, 1995.
- Yimin Huang and Marco Valtorta. Pearl’s calculus of intervention is complete. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI’06, pages 217–224. AUAI Press, 2006.
- Antti Hyttinen, Frederick Eberhardt, and Matti Järvisalo. Do-calculus when the true graph is unknown. In *UAI*, pages 395–404, 2015.
- Amin Jaber, Jiji Zhang, and Elias Bareinboim. Causal identification under Markov equivalence. Technical report, R-35, Purdue AI Lab, Department of Computer Science, Purdue University, 2018a.
- Amin Jaber, Jiji Zhang, and Elias Bareinboim. A graphical criterion for effect identification in equivalence classes of causal diagrams. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI’18, 2018b.
- Manabu Kuroki and Masami Miyakawa. Identifiability criteria for causal effects of joint interventions. *Journal of the Japan Statistical Society*, 29(2):105–117, 1999.
- Marloes H. Maathuis, Diego Colombo, Markus Kalisch, and Peter Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7(4):247–248, 2010.
- J. Pearl. Aspects of graphical models connected with causality. In *Proceedings of the 49th Session of the International Statistical Institute*, pages 391–401, Tome LV, Book 1, Florence, Italy, 1993.
- J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, 2000. 2nd edition, 2009.
- Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.
- Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic Books, 2018. forthcoming.
- Emilija Perković, Johannes Textor, Markus Kalisch, and Marloes H. Maathuis. A complete generalized adjustment criterion. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pages 682–691, 2015.
- Emilija Perković, Johannes Textor, Markus Kalisch, and Marloes H. Maathuis. Complete graphical characterization and construction of adjustment sets in Markov equivalence classes of ancestral graphs. *arXiv preprint arXiv:1606.06903*, 2016.
- Thomas Richardson and Peter Spirtes. Ancestral graph Markov models. *Annals of Statistics*, pages 962–1030, 2002.
- Thomas S Richardson, Robin J Evans, James M Robins, and Ilya Shpitser. Nested Markov properties for acyclic directed mixed graphs. *arXiv preprint arXiv:1701.06686*, 2017.
- 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.
- Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*, volume 81. MIT press, 2001.
- Jin Tian. *Studies in causal reasoning and learning*. PhD thesis, University of California, Los Angeles, 2002.
- Jin Tian and Judea Pearl. A general identification condition for causal effects. In *AAAI/IAAI*, pages 567–573, 2002.
- TS Verma. Graphical aspects of causal models. *Technical Report R-191, UCLA*, 1993.
- Jiji Zhang. *Causal inference and reasoning in causally insufficient systems*. PhD thesis, Carnegie Mellon University, 2006.
- Jiji Zhang. Generalized do-calculus with testable causal assumptions. In *International Conference on Artificial Intelligence and Statistics*, pages 667–674, 2007.
- Jiji Zhang. Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9(Jul):1437–1474, 2008a.
- Jiji 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, 2008b.