Constrained Identifiability of Causal Effects

Yizuo Chen[1], **Adnan Darwiche**[1]

[1]Computer Science Department, University of California, Los Angeles, USA
yizuo.chen@ucla.edu darwiche@cs.ucla.edu

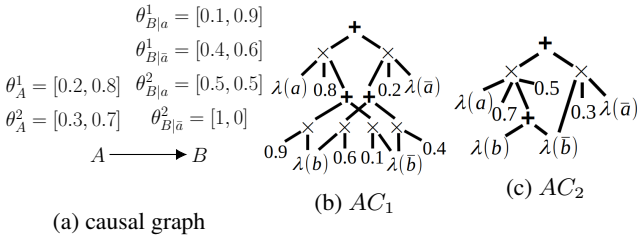
Abstract

We study the identification of causal effects in the presence of different types of constraints (e.g., logical constraints) in addition to the causal graph. These constraints impose restrictions on the models (parameterizations) induced by the causal graph, reducing the set of models considered by the identifiability problem. We formalize the notion of *constrained identifiability*, which takes a set of constraints as another input to the classical definition of identifiability. We then introduce a framework for testing constrained identifiability by employing tractable Arithmetic Circuits (ACs), which enables us to accommodate constraints systematically. We show that this AC-based approach is at least as complete as existing algorithms (e.g., do-calculus) for testing classical identifiability, which only assumes the constraint of strict positivity. We use examples to demonstrate the effectiveness of this AC-based approach by showing that unidentifiable causal effects may become identifiable under different types of constraints.

1 Introduction

A causal effect measures the impact of an intervention on an outcome of interest. For example, one may ask the question "how likely would an employee resign if the company reduced the bonus?" More generally, given a set of treatment variables $\mathbf{X}$ and another set of outcome variables $\mathbf{Y}$, the causal effect is defined as the probability of $\mathbf{y}$ under a treatment $do(\mathbf{x})$ and is commonly denoted by $\Pr(\mathbf{y}|do(\mathbf{x}))$ or $\Pr_{\mathbf{x}}(\mathbf{y})$. These queries belong to the second rung of Pearl's causal hierarchy and cannot be answered in general without conducting experiments (Pearl and Mackenzie 2018). When a causal graph is available, however, some of these causal effects may be answered from observational studies. This leads to the problem of causal-effect identifiability that studies whether a causal effect can be uniquely determined from observational distributions when a causal graph is given; see, e.g., (Pearl 2009; Imbens and Rubin 2015; Peters, Janzing, and Schölkopf 2017; Spirtes, Glymour, and Scheines 2000).

A recent line of research went beyond classical identifiability to show that more causal effects may become identifiable when additional information besides a causal graph is available. Works that fall into this line include the exploitation of the knowledge of context-specific independence (Boutilier et al. 1996; Tikka, Hyttinen, and Karvanen 2019) and more recently, functional dependencies (Chen and Darwiche 2024). In this work, we show that such knowledge can be understood as *constraints* on the models (parameterizations) induced by the causal graph. This leads us to formulate the notion of *constrained identifiability*,[1] which takes a set of models (defined by constraints) as an additional input to the classical definition of identifiability. Constrained identifiability is general enough to incorporate more types of constraints, such as fully-known observational distributions, as shown in this work. Since constraints reduce the distributions considered by identifiability, an unidentifiable causal effect may become identifiable due to available constraints.



(a) causal graph (b) AC_1 (c) AC_2

Figure 1: Arithmetic circuits for two different models.

We further propose an approach for testing constrained identifiability based on Arithmetic Circuits (AC) (Darwiche 2001, 2003, 2009, 2021b). In essence, an AC is a data structure representing a particular computation procedure and has played an influential role in probabilistic inference. In this work, we show yet another application of ACs in testing constrained identifiability based on the following observation: A causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable iff there exists an AC that computes $\Pr_{\mathbf{x}}(\mathbf{y})$ and whose output only depends on the observational distribution $\Pr(\mathbf{V})$. That is, if we can construct an AC to compute the causal effect and certify that the output of the AC is invariant under any fixed $\Pr(\mathbf{V})$, the causal effect is guaranteed to be identifiable. The construction of the AC can be done using existing knowledge compilation methods, which enables us to exploit parameter-specific constraints of the model such as equal parame-

[1]The notion was initially introduced in (Chen and Darwiche 2024) to address positivity assumptions (constraints). We generalize the notion here to incorporate any constraints.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

ters (e.g., context-specific independences) and 0/1 parameters (e.g., logical constraints) and yield simpler ACs; see, e.g., (Darwiche 2002; Chavira and Darwiche 2007; Huber et al. 2023; Huang and Darwiche 2024). Consider two different parameterizations Θ^1, Θ^2 for the same causal graph G in Figure 1a which contains two binary variables A, B. The AC for (G, Θ^2) in Figure 1c contains fewer nodes than the AC for (G, Θ^1) in Figure 1b since Θ^2 exhibits more constraints that get exploited when constructing the AC. As we will see later, constructing simpler ACs may lead to more complete causal-effect identifications using the method we shall propose. To check whether the output of an AC is invariant, we propose a method based on deriving an expression that is equivalent to the AC output and only involves the observed variables $\mathbf{V}$. We will show that this AC-based method can treat constraints in the form of context-specific independencies, functional dependencies, and fully-known observational distributions (that is, when $\Pr(\mathbf{V})$ is known).

This paper is structured as follows. We start with technical preliminaries on the problem of classical causal-effect identifiability in Section 2. We discuss the role of constraints in the context of identifiability and formally define the notion of constrained identifiability in Section 3. We introduce the AC-based method for testing constrained identifiability in Section 4 and propose a method for testing whether the output of an AC is invariant in Section 5. We demonstrate the effectiveness of AC-based method with examples in Section 6. We close with some concluding remarks in Section 7. All the proofs are included in the Appendix.

2 Technical Preliminaries

We consider discrete variables in this work. Single variables are denoted by uppercase letters (e.g., X) and their states are denoted by lowercase letters (e.g., x). Sets of variables are denoted by bold uppercase letters (e.g., $\mathbf{X}$) and their instantiations are denoted by bold lowercase letters (e.g., $\mathbf{x}$).

The problem of causal-effect identifiability studies whether a causal effect can be uniquely computed from a given causal graph G and a subset of observed variables $\mathbf{V}$. That is, to show that a causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable, we need to prove that all parameterizations for G that induce the same observational distribution $\Pr(\mathbf{V})$ also yield the same value for $\Pr_{\mathbf{x}}(\mathbf{y})$. While the general definition of identifiability (not necessarily for causal effects) in (Pearl 2009) does not restrict the observational distribution $\Pr(\mathbf{V})$, positivity assumptions (constraints) are often assumed to prevent zeros in $\Pr(\mathbf{V})$; see (Chen and Darwiche 2024; Kivva et al. 2022; Hwang et al. 2024) for recent discussions on positivity. The most common form of positivity constraints in the existing literature is strict positivity, i.e., $\Pr(\mathbf{V}) > 0$, which is a sufficient condition for complete causal-effect identification algorithms including the ID algorithm (Shpitser and Pearl 2006) and the do-calculus (Pearl 2009).[2] We will refer to identifiability with strict positivity as "classical identifiability," which is defined next.

[2](Chen and Darwiche 2024) showed that most causal effects are not identifiable unless we assume the positivity constraint $\Pr(X) > 0$ for each treatment variable X.

Definition 1 (Causal-Effect Identifiability)**.** *Let G be a casual graph and $\mathbf{V}$ be its observed variables. A causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is said to be identifiable with respect to $\langle G, \mathbf{V}\rangle$ if $\Pr^1_{\mathbf{x}}(\mathbf{y}) = \Pr^2_{\mathbf{x}}(\mathbf{y})$ for any pair of models M^1, M^2 that induce $\Pr^1, \Pr^2$ where $\Pr^1(\mathbf{V}) = \Pr^2(\mathbf{V}) > 0$.*

A causal effect is *unidentifiable* if it is not identifiable. To show that a causal effect is unidentifiable, it suffices to find two parametrizations for G that induce the same distribution $\Pr(\mathbf{V}) > 0$ yet different values for the causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$.

3 Constrained Identifiability

We next generalize this setup by considering more information, in addition to the causal graph and observed variables, as inputs to the identifiability problem. Past works that fit in this direction include the exploitation of *context-specific independences* in (Tikka, Hyttinen, and Karvanen 2019) and *functional dependencies* in (Chen and Darwiche 2024) to improve the identifiability of causal effects. The general setup we formulate in this work will allow us to also consider further information types such as *fully-known observational distributions* $\Pr(\mathbf{V})$, which can be readily available in practice when data on $\mathbf{V}$ allows us to estimate $\Pr(\mathbf{V})$.

Our setup is based on the notion of *constrained identifiability,* which was recently introduced in (Chen and Darwiche 2024) to systematically treat a variety of positivity constraints (assumptions). We will extend this notion next to incorporate arbitrary types of constraints.

We first define the (constrained) identifiability tuple. Here, we use "model" to mean a full parametrization of the causal graph (a model induces a distribution).

Definition 2. *We call $\langle G, \mathbf{V}, \mathcal{M}\rangle$ an identifiability tuple when G is a causal graph (DAG), $\mathbf{V}$ is its set of observed variables, and $\mathcal{M}$ is a set of models induced by G.*

The notion of constrained-identifiability can now be defined as follows. For simplicity, we will say "identifiability" to mean "constrained-identifiability" in the rest of paper.

Definition 3. *Let $\langle G, \mathbf{V}, \mathcal{M}\rangle$ be an identifiability tuple. A causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is said to be identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}\rangle$ if $\Pr^1_{\mathbf{x}}(\mathbf{y}) = \Pr^2_{\mathbf{x}}(\mathbf{y})$ for any pair of models $M^1, M^2 \in \mathcal{M}$ that induce $\Pr^1, \Pr^2$ where $\Pr^1(\mathbf{V}) = \Pr^2(\mathbf{V})$.*

Positivity constraints, context-specific independencies (CSI), and functional independencies restrict the set of models $\mathcal{M}$ in the definition of constrained-identifiability. That is, we only consider models that induce a distribution $\Pr$ that satisfies all the constraints above when testing for identifiability. To illustrate, consider the causal graph in Figure 3b where all variables are binary and X, Y are observed. If we impose the positivity constraint $\Pr(X, Y) > 0$, CSI constraint ($Y \perp\!\!\!\perp U|x$), and functional variable Y, then a model will be excluded from $\mathcal{M}$ if it assigns $Y \leftarrow X$ or $Y \leftarrow X \oplus U$ as a structural equation for Y.[3]

[3]The CSI constraint ($Y \perp\!\!\!\perp U|x$) says that Y is independent of U when $X = x$. Variable Y is functional means that Y is functionally determined by its parents (U and X). We will formally define these constraints in Section 6.

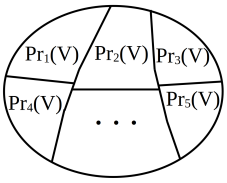
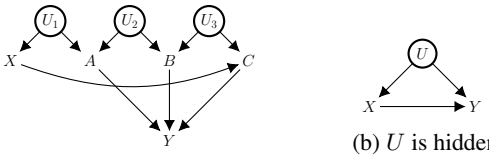


Figure 2: models for $\Pr(\mathbf{V})$



(a) U_1, U_2, U_3 are hidden (b) U is hidden

Figure 3: causal graphs that exhibit different behaviors of identifiability for $\Pr_x(y)$.

If $\mathcal{C}$ is a set of constraints, we will use $\mathcal{M}(\mathcal{C})$ to denote the set of models that satisfy $\mathcal{C}$. In the example above, we write the set of models as $\mathcal{M}[\Pr(X,Y) > 0, (Y \perp\!\!\!\perp U|x), (\mathbf{W} = \{Y\})]$, where $\mathbf{W}$ denotes variables with functional dependencies. When no constraint is imposed, we attain the weakest version of identifiability with $\mathcal{M}(\emptyset)$. Classical identifiability (Definition 1) corresponds to $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) > 0]\rangle$.

One type of constrained identifiability arises when we restrict the observational distribution to a fixed $\Pr^\star(\mathbf{V})$. That is, we only consider models that induce the given $\Pr^\star(\mathbf{V})$. To illustrate, consider the diagram in Figure 2, which contains the set of all models $\mathcal{M}$ induced by a causal graph. We can partition the set of models into *equivalence classes,* where all models in a particular class induce the same distribution $\Pr(\mathbf{V})$ — hence, an equivalence class is defined by its distribution $\Pr(\mathbf{V})$.[4] In classical identifiability, to show that a causal effect is identifiable, we need to show that all models in the same equivalence class produce the same causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$. When $\Pr^\star(\mathbf{V})$ is fixed, however, we only consider a single equivalence class $\Pr^\star(\mathbf{V})$ and check whether all models in this particular class produce the same value for the causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$. We denote the identifiability tuple under a fixed $\Pr^\star(\mathbf{V})$ as $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) = \Pr^\star(\mathbf{V})]\rangle$.

A causal effect is identifiable by Definition 1 iff it is identifiable under *all* fixed $\Pr^\star(\mathbf{V}) > 0$. However, a causal effect that is not identifiable by Definition 1 may still be identifiable under *some* $\Pr^\star(\mathbf{V}) > 0$. This happens when we have two distinct equivalence classes C_1 and C_2 where the models in C_1 agree on the causal effect but those in C_2 disagree. We next show a causal effect that belongs to this category.[5]

Proposition 1. *Consider the causal graph G in Figure 3a where $\mathbf{V} = \{X, A, B, C, Y\}$. The causal effect $\Pr_x(y)$ is unidentifiable wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) > 0]\rangle$ but is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) = \Pr^\star(\mathbf{V})]\rangle$ for some $\Pr^\star(\mathbf{V}) > 0$.*

The following causal effect is not identifiable for all $\Pr^\star(\mathbf{V}) > 0$ (it is not identifiable according to Definition 1).

Proposition 2. *Consider the causal graph G in Figure 3b where $\mathbf{V} = \{X, Y\}$. The causal effect $\Pr_x(y)$ is unidentifiable wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) = \Pr^\star(\mathbf{V})]\rangle$ for all $\Pr^\star(\mathbf{V}) > 0$.*

4 Testing Constrained-Identifiability Using Arithmetic Circuits

We next introduce a general method for testing constrained-identifiability based on Arithmetic Circuits (ACs) as defined in (Darwiche 2001, 2003, 2009). One main advantage of using ACs for testing identifiability is that more local structures such as the 0/1 parameters and equal parameters (including context-specific independences) can be exploited by the existing knowledge compilation methods (Darwiche 2002; Chavira and Darwiche 2005, 2007), leading to more complete causal-effect identifications under the additional constraints. Throughout the paper we will assume that the states of all variables (including $\mathbf{U}$) are known, and we will point out in case this assumption can be omitted.[6]

Our method for testing identifiability using ACs involves two steps (ingredients): *AC Construction* and *Invariance Testing*. The first step constructs an AC whose output computes the causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ and which incorporates the constraints. The second step tests identifiability by checking whether the output of the AC is invariant under all models that satisfy the constraints ($\mathcal{M}$ in Definition 3). Hence, the main objective of this and future work is to improve the quality of these two steps so that we obtain a more complete method for identifying causal effects under constraints.

4.1 AC Construction

We start by constructing an AC for the causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ based on the approach in (Darwiche 2021a). First, compile the original BN into an AC (in a standard way) that contains indicators λ_v for each variable state v and parameters $\theta_{v|\mathbf{p}}$ for each instantiation over V and its parents $\mathbf{P}$.[7] Second, replace every parameter $\theta_{x|\mathbf{p}}$ with 1 for each treatment $X \in \mathbf{X}$. Finally, for each $V \in \mathbf{X} \cup \mathbf{Y}$, assign $\lambda_v = 1$ if v is compatible with $\mathbf{x}, \mathbf{y}$ and $\lambda_v = 0$ otherwise; assign 1 to all other indicators.[8]

To evaluate an AC under a model M, we simply plugin each (non-constant) parameter $\theta_{v|\mathbf{p}}$ with the value from M and compute the output of the AC in a standard way. One key observation is that the ACs constructed from the above procedure are guaranteed to compute $\Pr_{\mathbf{x}}(\mathbf{y})$ correctly under *all* parameterizations of G. As we will see later, when additional constraints are available, the ACs may be further

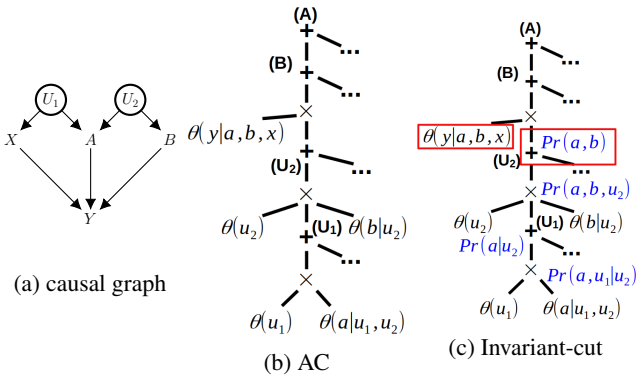
[4] There is an infinite number of such equivalence classes since a causal graph G induces infinitely many distributions $\Pr(\mathbf{V})$.

[5] The proof of Proposition 1 exploits the context-specific independence relations $(Y \perp\!\!\!\perp C \mid A = a, B)$ and $(Y \perp\!\!\!\perp B \mid A = \bar{a}, C)$; see (Tikka, Hyttinen, and Karvanen 2019) for more details about leveraging this type of constraints for identifiability.

[6] (Zhang, Tian, and Bareinboim 2022) showed that all counterfactual distributions (which subsumes observational and interventional distributions) can be captured by causal models whose hidden variables have a *bounded* number of states.

[7] Note that each $\theta_{v|\mathbf{p}}$ is equivalent to the conditional probability $\Pr(v|\mathbf{p})$ when $\Pr(\mathbf{p}) > 0$. Hence, we will use $\theta_{v|\mathbf{p}}$ and $\Pr(v|\mathbf{p})$ interchangeably when the positivity condition holds.

[8] The assignment is identical to the one used for AC estimation under evidence $\mathbf{x}, \mathbf{y}$.



(a) causal graph (b) AC (c) Invariant-cut

Figure 4: AC constructed for $\Pr_x(y)$ for a causal graph; only one branch for each $+$-node is plotted; equivalent expressions are shown in blue and nodes in the invariant-cut are marked with red boxes.

simplified, leading to more complete causal-effect identifiability results in practice. In general, the leaves of an AC are either constants (e.g., 0.8) or parameters ($\theta_{v|\mathbf{p}}$).

Consider the causal graph G in Figure 4a with observed variables $\mathbf{V} = \{X, A, B, Y\}$. Assuming all variables are binary, Figure 4b depicts an AC for $\Pr_x(y)$. Due to the space limit, we only plotted one branch for most $+$-nodes in the figure. In general, we prefer simpler (smaller) ACs since they can sometimes lead to more complete causal-effect identifiability. While most AC simplifications can be handled automatically by the ACE package,[9] we propose additional rules that can be used to further simplify ACs in certain scenarios; see Appendix A for more details.

4.2 Invariance Testing

The AC constructed from the previous step is guaranteed to compute the causal effect under all models in $\mathcal{M}$. We next develop a method for testing identifiability based on the following observation: a causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}\rangle$ if the output of the ACs for $\Pr_{\mathbf{x}}(\mathbf{y})$ is *invariant* under all models (in $\mathcal{M}$) that induce a same $\Pr(\mathbf{V})$.

In this work, an *expression* is defined as a composition of constants (e.g., 0.8), parameters ($\theta_{v|\mathbf{p}}$), and conditional probabilities (e.g., $\Pr(a|b)$). Note that each parameter $\theta_{v|\mathbf{p}}$ is equal to the conditional probability $\Pr(v|\mathbf{p})$ when $\Pr(\mathbf{p}) > 0$. For example, $\Pr(y|x)$, 0.65, $0.2\Pr(a|x) + 0.6\Pr(x)$, $\sum_u \Pr(a|x)\Pr(u)$, $\theta(a|x)\Pr(x) + 0.3\Pr(\bar{a})$ are all valid expressions. An AC specifies an expression consisting of $\times$'s and $+$'s, which can be obtained by evaluating the output of the AC in a standard (bottom-up) way. For example, the AC in Figure 4b specifies the expression $\sum_a \sum_b \theta_{y|a,b,x} \sum_{u_2} \theta_{u_2}\theta_{b|u_2} \sum_{u_1} \theta_{u_1}\theta_{a|u_1,u_2}$. Moreover, every node in an AC corresponds to an expression so we will use "AC node" and "expression" interchangeably.

Our next goal is to propose a method for testing identifiability that operates on ACs. We start by defining some key notions that are required to achieve this goal.

[9] http://reasoning.cs.ucla.edu/ace.

Definition 4. *An expression is called $\langle G, \mathbf{V}, \mathcal{M}\rangle$-invariant if it evaluates to the same value for all models in $\mathcal{M}$ that induce a same $\Pr(\mathbf{V})$.*

The following result shows that we can test identifiability by checking whether the output of an AC is invariant.

Proposition 3. *A causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}\rangle$ iff there is an AC for $\Pr_{\mathbf{x}}(\mathbf{y})$ whose output expression is $\langle G, \mathbf{V}, \mathcal{M}\rangle$-invariant.*

In general, it is difficult to tell whether the output of an AC is invariant since the evaluation of the AC often contains hidden variables $\mathbf{U}$. However, if we are able to replace the expressions for some AC nodes with *equivalent expressions* that do not depend on $\mathbf{U}$, we may obtain output expressions that are obviously invariant, e.g., expressions that only involve observed variables $\mathbf{V}$.

Definition 5. *Two expressions $\mathcal{E}_1$ and $\mathcal{E}_2$ are called $\langle G, \mathbf{V}, \mathcal{M}\rangle$-equivalent if they evaluate to the same value under all models $\mathcal{M}$. If in addition $\mathcal{E}_1$ does not contain any parameters (θ's)[10] and does not mention any hidden variables, then $\mathcal{E}_1$ is called an $\langle G, \mathbf{V}, \mathcal{M}\rangle$-projection of $\mathcal{E}_2$.*

The following theorem presents a method for testing identifiability.

Proposition 4. *The causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}\rangle$ iff there is an AC for $\Pr_{\mathbf{x}}(\mathbf{y})$ that satisfies the following condition: there is a cut between the root and leaf nodes where every node on the cut has a $\langle G, \mathbf{V}, \mathcal{M}\rangle$-projection.*

We call the cut in Proposition 4 an *invariant-cut*. Figure 4c depicts the projections (in blue) and an invariant-cut (in red boxes) for the AC in Figure 4b. Hence, we conclude that the causal effect $\Pr_x(y)$ is identifiable. Once we attain an invariant-cut we immediately obtain an identifying formula by plugging in the projections for the nodes on the invariant-cut and evaluating the AC root. In this example, the identifying formula for the causal effect is $\Pr_x(y) = \Pr(y|x,a,b)\Pr(a,b) + \Pr(y|x,a,\bar{b})\Pr(a,\bar{b}) + \Pr(y|x,\bar{a},b)\Pr(\bar{a},b) + \Pr(y|x,\bar{a},\bar{b})\Pr(\bar{a},\bar{b})$, which resembles the backdoor adjustment formula. Invariant-cuts may not be unique as the choice of projections is not unique. We may also have multiple ACs with invariant-cuts, which further contributes to the multiplicity of identifying formulas.

Another interesting observation on Proposition 4 is that the root of the AC always constitutes a valid invariant-cut if the causal effect is identifiable. Consider the following expression (ratio of two infinite series) for the root: $\frac{\sum_{M\in\mathcal{M}} \mathbb{1}\{\Pr(\mathbf{V})=\Pr^M(\mathbf{V})\}\Pr^M_{\mathbf{x}}(\mathbf{y})}{\sum_{M\in\mathcal{M}} \mathbb{1}\{\Pr(\mathbf{V})=\Pr^M(\mathbf{V})\}}$ where $\Pr^M$ denotes the distribution induced by a specific model M. If $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable, the expression is a valid identifying formula for

[10] The observational distribution $\Pr(\mathbf{V})$ does not depend on the value of parameter $\theta_{x|\mathbf{p}}$ when $\Pr(\mathbf{p}) = 0$ — but the interventional distribution $\Pr_{\mathbf{x}}(.)$ and, hence, causal effects may depend on the value of $\theta_{x|\mathbf{p}}$ in this case. This is why we require projections to be free of parameters as they may not be computable from $\Pr(\mathbf{V})$. However, since $\theta_{x|\mathbf{p}} = \Pr(x|\mathbf{p})$ when $\Pr(\mathbf{p}) > 0$ we can always replace $\theta_{x|\mathbf{p}}$ with $\Pr(x|\mathbf{p})$ when $\Pr(\mathbf{p}) > 0$.

computing $\Pr_{\mathbf{x}}(\mathbf{y})$ since every model M that induces $\Pr(\mathbf{V})$ also induces a same $\Pr^M_{\mathbf{x}}(\mathbf{y})$. Hence, the root of the AC always constitutes an invariant-cut in this case since any identifying formula constitutes a projection for the root. However, since lower cuts involve smaller expressions, finding efficient projections for nodes on such cuts may be easier.

5 Finding Invariant-Cuts For ACs

Developing a complete method for finding invariant-cuts, if they exist, is beyond the scope of this paper as it represents a first step in the proposed direction. However, we will provide a complete method in this section assuming we only have the classical positivity constraint, $\Pr(\mathbf{V}) > 0$. This case has already been solved in the literature using methods such as the ID algorithm and the do-calculus, so the results in this section provide another way of solving this case. These results also show that the proposed framework in this paper is at least as powerful as these classical methods. Moreover, we will use these results in the next section where we show how to handle other types of constraints but in a less complete fashion as far as identifying invariant-cuts.

Our method for finding projections for ACs is based on the notion of c-components, which has been studied extensively in (Tian and Pearl 2002, 2003; Shpitser and Pearl 2006; Huang and Valtorta 2006). Given a causal graph G, a (maximal) c-component is a (maximal) subgraph of G in which all variables are connected by bidirected paths,[11] and each c-component S is associated with a *c-factor* defined as $Q[S](\mathbf{V}) = \sum_{\mathbf{U}} \prod_{U \in \mathbf{U}} \theta_U \prod_{V \in \mathbf{V}} \theta_{V|\mathbf{P}_V}$ where $\mathbf{U}, \mathbf{V}, \mathbf{P}_V$ denote the hidden variables, observed variables, and parents of V. Moreover, the causal graph G can be decomposed into c-components $\mathcal{S} = \{S_1, \ldots, S_k\}$ such that $\Pr(\mathbf{V}) = \prod_{S \in \mathcal{S}} Q[S]$ when $\Pr(\mathbf{V}) > 0$. Existing methods in (Tian and Pearl 2003; Shpitser and Pearl 2006; Huang and Valtorta 2006) identify causal effects by decomposing the mutilated graph G' (under $do(\mathbf{x})$) into c-components $\mathcal{S}$ and then checking whether each $Q[S]$ (for $S \in \mathcal{S}$) has a projection which only involves $\mathbf{V}$.[12] We next present a notion that summarizes such c-components, which follows from the key results in (Tian and Pearl 2003; Huang and Valtorta 2006; Shpitser and Pearl 2006) and is defined in an inductive way.

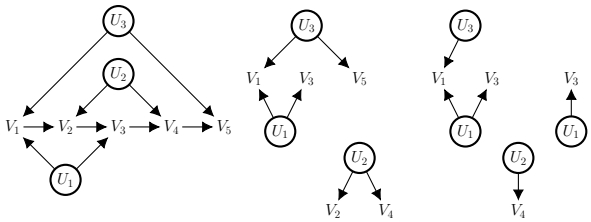
Definition 6. *Let G be a causal graph with observed variables $\mathbf{V}$. A subgraph S of G is $\mathbf{V}$-computable if one of the following holds: (1) $S = G$; (2) S is a maximal c-component of a $\mathbf{V}$-computable subgraph; or (3) S is the result of pruning a leaf node from a $\mathbf{V}$-computable subgraph.*

We can always find a projection for each $Q[S]$ as stated by the following corollary.

Corollary 1. *Let G be a causal graph with observed variables $\mathbf{V}$. If S is a $\mathbf{V}$-computable subgraph of G, then $Q[S]$ has a $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) > 0]\rangle$-projection.*

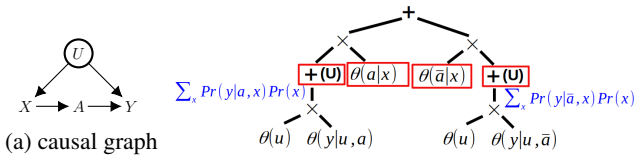
[11] A bidirected path is a path that consists of bidirected edges in the form of $X \leftarrow U \rightarrow Y$. We assume the causal graph G is semi-Markovian here; that is, all hidden variables are roots in G and have exactly two children. A more general definition of c-components beyond semi-Markovian can be found in (Tian and Pearl 2002).

[12] A mutilated graph under $do(\mathbf{x})$ is attained by removing all the incoming edges for $\mathbf{X}$ from the original causal graph.



(a) causal graph (b) $\mathbf{V}$-computable subgraphs

Figure 5: $\mathbf{V}$-computable subgraphs for a causal graph from (Tian and Pearl 2002).



(a) causal graph (b) projections and an invariant-cut

Figure 6: An AC constructed for $\Pr_x(y)$; an invariant-cut (in red boxes) is found by the c-component method.

To illustrate, Figure 5b depicts some $\mathbf{V}$-computable subgraphs for the causal graph in Figure 5a. The upper-left subgraph S in Figure 5b corresponds to the following c-factor: $Q[S] = \sum_{U_1,U_3} \theta_{U_1}\, \theta_{U_3}\, \theta_{V_1|U_1,U_3}\, \theta_{V_3|U_1,V_2}\, \theta_{V_5|U_3,V_4}$. Now suppose the expression at some AC node i matches the c-factor for a $\mathbf{V}$-computable subgraph, we can replace it with the corresponding projection which only involves $\mathbf{V}$; see Appendix B for more details.

Consider the causal graph in Figure 6a with binary observed variables $\mathbf{V} = \{X, Y, A\}$ and assume $\Pr(\mathbf{V}) > 0$. Figure 6b depicts an AC constructed for the causal effect $\Pr_x(y)$. If we consider the $\mathbf{V}$-computable subgraph S that contains $U \rightarrow Y$, we obtain the following c-factor $Q[S] = \sum_U \theta_U \theta_{Y|U,A}$ for S, which matches exactly the expression at the $+$-node on the left branch of the AC. Hence, we can assign the corresponding projection for the $+$-node as shown in Figure 6b. It turns out that the causal effect is identifiable since we can find an invariant-cut (in red boxed) for the AC.

The following theorem shows that the *c-component method*, when coupled with carefully designed ACs, is sound and complete for testing identifiability under $\Pr(\mathbf{V}) > 0$ like the ID algorithm (Shpitser and Pearl 2006) and do-calculus (Pearl 2009).

Theorem 1. *A causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable (wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) > 0]\rangle$) iff there is an AC for $\Pr_{\mathbf{x}}(\mathbf{y})$ on which the c-component method finds an invariant-cut.*[13]

6 Exploiting Constraints In Causal-Effect Identifiability: Examples

We next use examples to demonstrate how our AC-based method can be applied to improve causal-effect identifiability under different types of constraints, some of which

[13] We assume all variables are binary when constructing the AC.

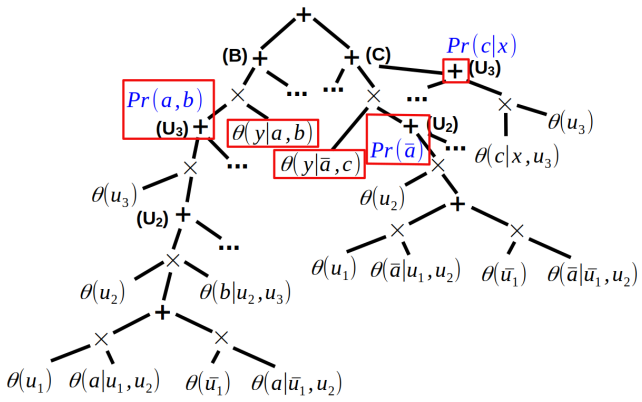


Figure 7: AC for Figure 3a with projections derived with the c-component method; projections are marked near the nodes in blue; the invariant-cut is marked with red boxes.

have been treated using methods dedicated to such constraints. In particular, we consider three types of constraints: (1) context-specific independencies; (2) functional dependencies; and (3) fully-known $\Pr(\mathbf{V})$, and show that we can exploit these constraints to simplify the constructed ACs in order to facilitate the discovery of invariant-cuts.

All the ACs considered in this section are compiled using the variable elimination method discussed in (Chavira and Darwiche 2007), with constrained variable elimination orders: first eliminate all hidden variables $\mathbf{U}$, then observed variables $\mathbf{V}$ in a bottom-up order. These ACs may be further simplified using the simplification rules in Appendix A.

6.1 Context-Specific Independence

We first illustrate that the AC-based method can improve identifiability given knowledge of context-specific independence (CSIs) (Boutilier et al. 1996; Tikka, Hyttinen, and Karvanen 2019; Shen, Choi, and Darwiche 2020). CSI has the form $(Y \perp\!\!\!\perp X|\mathbf{p})$ where $X, \mathbf{P}$ are the parents of Y. That is, Y is independent of X when the values of other parents $\mathbf{P}$ are set to $\mathbf{p}$. For simplicity, we write $(Y \perp\!\!\!\perp X|\mathbf{P})$ if CSI holds under all instantiations of $\mathbf{P}$.

CSI constraints can be incorporated into the construction of ACs as follows. For each CSI relation $(Y \perp\!\!\!\perp X|\mathbf{p})$, we merge all the parameters $\theta_{y|x,\mathbf{p}}$ (which are leaves) in the AC into a single node $\theta_{y|\mathbf{p}}$. This is valid since $\theta_{y|x,\mathbf{p}} = \Pr(y|\mathbf{p})$ for all x and y (Darwiche 2002). We can then apply simplification rules, such as those provided in Appendix A, to facilitate the discovery of invariant-cuts.

Consider again the causal graph in Figure 3a where the causal effect $\Pr_x(y)$ is not identifiable under the positivity constraint $\Pr(\mathbf{V}) > 0$. Assume now the following CSI constraints $\mathcal{C}_{csi}$: $(Y \perp\!\!\!\perp C|a, B)$ and $(Y \perp\!\!\!\perp B|\bar{a}, C)$. We can construct the simplified AC in Figure 7 using the procedure above. We can then find an invariant-cut for the AC using the c-component method; see Appendix C.1 for the derivation. Hence, we conclude that the causal effect is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}[\mathcal{C}_{csi}, \Pr(\mathbf{V}) > 0]\rangle$. We show another example adapted from (Tikka, Hyttinen, and Karvanen 2019) in Appendix C.2 to further illustrate the advantages brought by CSI constraints to identification, where our AC-based method yields the same identifying formula as the one found by the method in (Tikka, Hyttinen, and Karvanen 2019).

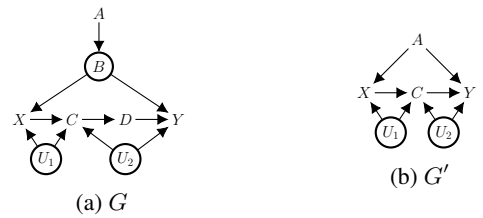


Figure 8: A, C, D, X, Y are observed; B, D are functional.

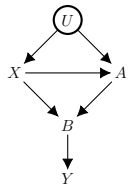
6.2 Functional Dependencies

In a causal graph, a variable W with parents $\mathbf{P}$ is said to be *functional,* or exhibit a *functional dependency,* if $\theta_{w|\mathbf{p}} \in \{0, 1\}$ for all instantiations $w, \mathbf{p}$ — we assume that we do not know the specific values for these $\theta_{w|\mathbf{p}}$. Functional dependencies have been recently exploited to improve the identifiability of causal effects (Chen and Darwiche 2024) and the computation of causal queries (Darwiche 2020; Chen and Darwiche 2022; Han, Chen, and Darwiche 2023).

Let $\mathbf{W}$ be a subset of variables that exhibit functional dependencies. (Chen and Darwiche 2024) showed that an unidentifiable causal effect may become identifiable given the functional variables $\mathbf{W}$, without the need of knowing the specific functions ($\theta_{w|\mathbf{p}}$'s) for $\mathbf{W}$. In addition, the work introduced a reduction-based method for testing identifiability under functional dependencies by eliminating (removing) a subset of functional variables $\mathbf{Q} \subseteq \mathbf{W}$ from the causal graph.[14] That is, a causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable wrt $\langle G, \mathbf{V}, \mathcal{M}\rangle$ iff it is identifiable wrt $\langle G', \mathbf{V}', \mathcal{M}'\rangle$, where $\mathbf{V}' = \mathbf{V} \setminus \mathbf{Q}$, G' is the result of eliminating $\mathbf{Q}$ from G, and $\mathcal{M}'$ is obtained from $\mathcal{M}$ by changing functional variables $\mathbf{W}$ to $(\mathbf{W} \setminus \mathbf{Q})$. Hence, we can use the elimination of functional variables as a *preprocessing mechanism* to reduce the identifiability problem with functional dependencies into classical identifiability (without functional dependencies).

When the states of variables are known, the if-direction of the result still holds. If we can show that $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable wrt $\langle G', \mathbf{V}', \mathcal{M}'\rangle$ using the AC-based method, the same causal effect is guaranteed to be identifiable wrt the original $\langle G, \mathbf{V}, \mathcal{M}\rangle$ with functional dependencies. We next illustrate this with an example. Consider the causal graph G in Figure 8a with observed variables $\mathbf{V} = \{A, C, D, X, Y\}$ and the causal effect $\Pr_x(y)$ wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(A, C, X, Y) > 0, (C \perp\!\!\!\perp U_1|x, U_2), \mathbf{W} = \{B, D\}]\rangle$. We first apply the preprocessing mechanism to eliminate the functional variables, which yields the causal graph G' in Figure 8b. Since there is no functional variables in G', we can now apply the

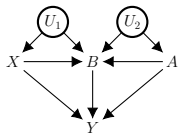
[14] The result holds if every $Q \in \mathbf{Q}$ satisfies the following condition: $Q \notin \mathbf{X} \cup \mathbf{Y}$ and every path from some hidden, non-functional variable to Q contains an observed variable (other than Q). See (Chen and Darwiche 2024) for details about the elimination operation and its required positivity constraints.



(a) causal graph

$\Pr((b_1, b_2, b_3, b_4)|x, a) = (0.3, 0.3, 0.1, 0.3)$
$\Pr((b_1, b_2, b_3, b_4)|x, \bar{a}) = (0.3, 0.3, 0.2, 0.2)$
$\Pr(y|b_1) = 0.2, \Pr(y|b_2) = 0.6$
$\Pr(y|b_3) = \Pr(y|b_4) = 0.1$

(b) conditional probabilities in $\Pr^\star(\mathbf{V})$



(c) causal graph

$\Pr((y_1, y_2, y_3)|x, a, b) = (0, 0.4, 0.6)$
$\Pr((y_1, y_2, y_3)|x, a, \bar{b}) = (0, 0.3, 0.7)$
$\Pr((y_1, y_2, y_3)|x, \bar{a}, b) = (0.2, 0.1, 0.7)$
$\Pr((y_1, y_2, y_3)|x, \bar{a}, \bar{b}) = (0.5, 0.2, 0.3)$
$(B \perp\!\!\!\perp U_1|x, \bar{a}, U_2)$

(d) conditional probabilities in $\Pr^\star(\mathbf{V})$ and a CSI constraint.

Figure 9: Examples for fully-known $\Pr(\mathbf{V})$.

AC-based method and show that $\Pr_x(y)$ is identifiable wrt $\langle G', \mathbf{V}' = \{A, C, X, Y\}, \mathcal{M}'[\Pr(A, C, X, Y) > 0, (C \perp\!\!\!\perp U_1|x, U_2)]\rangle$; see Appendix C.3 for a detailed derivation. This implies that the causal effect is identifiable wrt the original identifiability tuple (with functional dependencies). Moreover, this example shows how the proposed framework can treat a combination of constraint types, in the form of CSI and functional dependencies.

6.3 Fully-Known $\Pr(\mathbf{V})$

When an observational distribution $\Pr^\star(\mathbf{V})$ is given in addition to the causal graph G, we have an identifiability problem (wrt $\langle G, \mathbf{V}, \mathcal{M}[\Pr(\mathbf{V}) = \Pr^\star(\mathbf{V})]\rangle$) as defined previously. This is quite common in practice when data over observed variables $\mathbf{V}$ allows us to estimate $\Pr^\star(\mathbf{V})$ accurately.

Knowing $\Pr^\star(\mathbf{V})$ leads to several advantages. First, the positivity constraint over $\Pr^\star(\mathbf{V})$ becomes immediate since all the zero entries are determined by $\Pr^\star(\mathbf{V})$. Second, all the (well-defined) parameters $\theta_{v|\mathbf{p}}$ over observed variables $V, \mathbf{P}$ can be fixed to constant values in the AC. This allows us to exploit more parameter-specific knowledge such as 0/1 parameters and equal parameters (subsumes CSI) as exhibited by $\Pr^\star(\mathbf{V})$. One key observation is that a causal effect $\Pr_{\mathbf{x}}(\mathbf{y})$ is identifiable under a known $\Pr^\star(\mathbf{V})$ iff $\Pr_{\mathbf{x}}(\mathbf{y})$ always evaluates to a *fixed constant* under all models that induce $\Pr^\star(\mathbf{V})$. This result follows directly from Proposition 3. In fact, the constant can always be evaluated by computing $\Pr_{\mathbf{x}}(\mathbf{y})$ under any single model that induces $\Pr^\star(\mathbf{V})$.

Consider the causal graph in Figure 9a where variable B has states b_1, b_2, b_3, b_4 and all other variables are binary. Suppose $\Pr^\star(\mathbf{V}) > 0$ satisfies the conditional probabilities in Figure 9b. We show that the causal effect $\Pr_x(y)$ is identifiable under the fixed $\Pr^\star(\mathbf{V}) > 0$ (this causal effect is not identifiable if $\Pr(\mathbf{V})$ is not fixed to $\Pr^\star(\mathbf{V})$). Figure 10 depicts the AC for the causal effect after plugging in the known constant values for parameters. If we further apply simplification rules on the AC, we obtain an AC which contains a single leaf with constant 0.28; see Appendix C.4 for more details. That is, the causal effect $\Pr_x(y) = 0.28$ for all models that induce $\Pr^\star(\mathbf{V})$ and is thus identifiable.

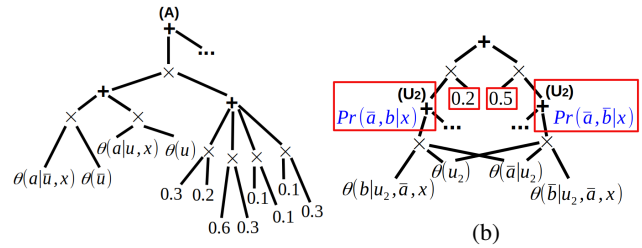


Figure 10: ACs constructed for Figure 9a and Figure 9c under known $\Pr^\star(\mathbf{V})$.

The constraint of known $\Pr(\mathbf{V})$ can be combined with other types of constraints to further improve the identifiability of causal effects. We next show an example that assumes both known $\Pr(\mathbf{V})$ and context-specific independences. Consider the causal graph in Figure 9c where variable Y has three states y_1, y_2, y_3 and all other variables are binary. Suppose the fixed $\Pr^\star(\mathbf{V})$ satisfies the conditional probabilities in Figure 9d and the CSI constraint $(B \perp\!\!\!\perp U_1|x, \bar{a}, U_2)$. We next show that the causal effect $\Pr_x(y_1)$ is identifiable under these constraints. Note that strict positivity no longer holds in this example due to the zero entries in the CPT of Y. In fact, the existence of zero entries enables the identifiability since it allows us to prune nodes that evaluate to zero from the AC. Figure 10b depicts the AC for $\Pr_x(y_1)$ after applying the simplification rules in Appendix A along with the projections and invariant-cut. Similar to the examples in Figure 8 and Appendix C.2, we need to modify the c-component method slightly to find the projections; see Appendix C.5 for details. The identifying formula in this case is $\Pr_x(y_1) = 0.2\Pr(\bar{a}, b|x) + 0.5\Pr(\bar{a}, \bar{b}|x)$ which can be evaluated using the known $\Pr^\star(\mathbf{V})$.

7 Conclusion

We formalized the notion of constrained identifiability which targets causal-effect identifiability in the presence of various types of constraints. We also proposed an approach for testing constrained identifiability based on constructing an Arithmetic Circuit (AC) for computing the causal effect using the causal graph and available constraints, and then finding invariant-cuts. We showed that this method is at least as complete as classical methods such as the ID algorithm and the do-calculus (which handle strict positivity constraints). We further demonstrated with examples how this approach can be used for testing causal-effect identifiability under various types of constraints, including context-specific independencies, functional dependencies, and fully-known observational distributions, therefore improving the completeness of classical methods when going beyond strict positivity constraints.

References

Boutilier, C.; Friedman, N.; Goldszmidt, M.; and Koller, D. 1996. Context-Specific Independence in Bayesian Networks. In *UAI*, 115–123. Morgan Kaufmann.

Chavira, M.; and Darwiche, A. 2005. Compiling Bayesian Networks with Local Structure. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*, 1306–1312.

Chavira, M.; and Darwiche, A. 2007. Compiling Bayesian Networks Using Variable Elimination. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2443–2449.

Chen, Y.; and Darwiche, A. 2022. On the Definition and Computation of Causal Treewidth. In *UAI, Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence*.

Chen, Y.; and Darwiche, A. 2024. Identifying Causal Effects Under Functional Dependencies. In *NeurIPS*.

Darwiche, A. 2001. Decomposable negation normal form. *J. ACM*, 48(4): 608–647.

Darwiche, A. 2002. A Logical Approach to Factoring Belief Networks. In *KR*, 409–420. Morgan Kaufmann.

Darwiche, A. 2003. A differential approach to inference in Bayesian networks. *J. ACM*, 50(3): 280–305.

Darwiche, A. 2009. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press.

Darwiche, A. 2020. An Advance on Variable Elimination with Applications to Tensor-Based Computation. In *ECAI*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, 2559–2568. IOS Press.

Darwiche, A. 2021a. Causal Inference with Tractable Circuits. In *WHY Workshop, NeurIPS*. Https://arxiv.org/abs/2202.02891.

Darwiche, A. 2021b. Tractable Boolean and Arithmetic Circuits. In *Neuro-Symbolic Artificial Intelligence*, volume 342 of *Frontiers in Artificial Intelligence and Applications*, 146–172. IOS Press.

Han, Y.; Chen, Y.; and Darwiche, A. 2023. On the Complexity of Counterfactual Reasoning. In *IJCAI*, 5676–5684. ijcai.org.

Huang, H.; and Darwiche, A. 2024. Causal Unit Selection using Tractable Arithmetic Circuits. In *FLAIRS*. AAAI Press.

Huang, Y.; and Valtorta, M. 2006. Identifiability in Causal Bayesian Networks: A Sound and Complete Algorithm. In *AAAI*, 1149–1154. AAAI Press.

Huber, D.; Chen, Y.; Antonucci, A.; Darwiche, A.; and Zaffalon, M. 2023. Tractable Bounding of Counterfactual Queries by Knowledge Compilation. *Sixth Workshop on Tractable Probabilistic Modeling @ UAI 2023*.

Hwang, I.; Choe, Y.; Kwon, Y.; and Lee, S. 2024. On Positivity Condition for Causal Inference. In *ICML*. OpenReview.net.

Imbens, G. W.; and Rubin, D. B. 2015. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press.

Kivva, Y.; Mokhtarian, E.; Etesami, J.; and Kiyavash, N. 2022. Revisiting the general identifiability problem. In *UAI*, volume 180 of *Proceedings of Machine Learning Research*, 1022–1030. PMLR.

Pearl, J. 2009. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, second edition.

Pearl, J.; and Mackenzie, D. 2018. *The Book of Why: The New Science of Cause and Effect*. Basic Books.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press.

Shen, Y.; Choi, A.; and Darwiche, A. 2020. A New Perspective on Learning Context-Specific Independence. In *PGM*, volume 138 of *Proceedings of Machine Learning Research*, 425–436. PMLR.

Shpitser, I.; and Pearl, J. 2006. Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models. In *AAAI*, 1219–1226. AAAI Press.

Spirtes, P.; Glymour, C.; and Scheines, R. 2000. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press. ISBN 978-0-262-19440-2.

Tian, J.; and Pearl, J. 2002. On the Testable Implications of Causal Models with Hidden Variables. In *UAI*, 519–527. Morgan Kaufmann.

Tian, J.; and Pearl, J. 2003. On the Identification of Causal Effects. *Technical Report*, R-290-L.

Tikka, S.; Hyttinen, A.; and Karvanen, J. 2019. Identifying Causal Effects via Context-specific Independence Relations. In *NeurIPS*, 2800–2810.

Zhang, J.; Tian, J.; and Bareinboim, E. 2022. Partial Counterfactual Identification from Observational and Experimental Data. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, 26548–26558. PMLR.

A Simplifying Arithmetic Circuits

The constructed AC can be simplified (reduced) by exploiting parameter-specific information such as 0/1 entries and equal parameters. While most of the simplifications can be handled by the ACE package,[15] we introduce three more rules that may lead to additional simplifications:

1. (sum-out) For a $+$-node in the AC, if it computes $\sum_z \Pr(z|\mathbf{w})$, replace the $+$-node with constant 1.
2. (merge) If two nodes in the AC are same and have a same set of children, merge them into a single node.
3. (distributivity) For a $+$-node in the AC, if it computes $\sum_i (c \cdot s_i)$, simplify it to $c \cdot \sum_i s_i$.

These rules can be applied recursively to simplify the AC to the maximum extent. Note that we can always omit AC nodes that evaluate to zero and nodes that evaluate to one if their parent is a $\times$-node. Please check out examples and their derivations in Appendix C for more details.

B C-component Method For Finding Projections

The c-component method finds projections by decomposing a causal graph into (maximal) c-components (Tian and Pearl 2002, 2003; Shpitser and Pearl 2006).[16] As shown in the main paper, a causal graph induces a set of $\mathbf{V}$-computable subgraphs. Moreover, each $\mathbf{V}$-computable subgraph S induces a c-factor (Tian and Pearl 2002, 2003; Huang and Valtorta 2006) which we denote as $Q[S]$. For example, the $\mathbf{V}$-computable subgraphs in Figure 5b correspond to the following formulas:

- $Q[S_1] = \sum_{U_1,U_3} \theta_{U_1}\theta_{U_3}\theta_{V_1|U_1,U_3}\theta_{V_3|U_1,V_2}\theta_{V_5|U_3,V_4}$
- $Q[S_2] = \sum_{U_1,U_3} \theta_{U_1}\theta_{U_3}\theta_{V_1|U_1,U_3}\theta_{V_3|U_1,V_2}$
- $Q[S_3] = \sum_{U_1} \theta_{U_1}\theta_{V_3|U_1,V_2}$
- $Q[S_4] = \sum_{U_2} \theta_{U_2}\theta_{V_2|U_2,V_1}\theta_{V_4|U_2,V_3}$
- $Q[S_5] = \sum_{U_2} \theta_{U_2}\theta_{V_4|U_2,V_3}$

To see why the c-factors for $\mathbf{V}$-computable subgraphs can be computed from $\Pr(\mathbf{V})$, consider the following inductive argument. For the original causal graph G, $Q[G] = \Pr(\mathbf{V})$ which is a projection. Whenever we prune a leaf node W from a $\mathbf{V}$-computable graph G to obtain a new graph G' (case (3) in Definition 6), the c-factor for G' is equal to $Q[G'] = \sum_W Q[G]$, which has a projection since $Q[G]$ has a projection. Whenever we decompose a $\mathbf{V}$-computable subgraph G into c-components $\mathcal{S} = \{S_1, \ldots, S_k\}$ (case (2) in Definition 6), we can compute each $Q[S]$ for each $S \in \mathcal{S}$ from $Q[G]$ using the "Generalized Q-decomposition" in (Tian and Pearl 2003). Hence, each $Q[S]$ has a projection since $Q[G]$ has a projection.

The projections for the c-factors of $\mathbf{V}$-computable subgraphs in Figure 5b are shown below, where $\equiv$ denotes the equivalence between the c-factor and projection.

[15] http://reasoning.cs.ucla.edu/ace.

[16] Any causal graph can be decomposed into c-components as shown in (Tian and Pearl 2002, 2003), though it is common to consider semi-Markovian models in which all hidden variables are roots and have two children.

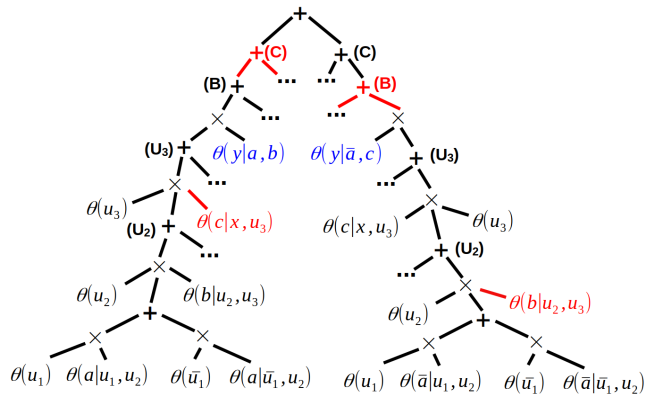


Figure 11: Original AC compiled from Figure 3a under CSI constraints $(Y \perp\!\!\!\perp C|A = a, B)$ and $(Y \perp\!\!\!\perp B|A = \bar{a}, C)$.

- $Q[S_1] \equiv \Pr(V_1)\Pr(V_3|V_1,V_2)\Pr(V_5|V_1,V_2,V_3,V_4)$
- $Q[S_2] \equiv \Pr(V_1)\Pr(V_3|V_1,V_2)$
- $Q[S_3] \equiv \sum_{V_1}\Pr(V_1)\Pr(V_3|V_1,V_2)$
- $Q[S_4] \equiv \Pr(V_2|V_1)\Pr(V_4|V_1,V_2,V_3)$
- $Q[S_5] \equiv \sum_{V_2}\Pr(V_2|V_1)\Pr(V_4|V_1,V_2,V_3)$

C Examples and Derivations

C.1 Derivation For Figure 7

Figure 11 depicts the original AC for $\Pr_x(y)$ under the CSI constraints after merging $\theta_{y|a,b,c}$ and $\theta_{y|a,b,\bar{c}}$ into $\theta_{y|a,b}$, and merging $\theta_{y|\bar{a},b,c}$ and $\theta_{y|\bar{a},\bar{b},c}$ into $\theta_{y|\bar{a},c}$ (merged nodes are color-coded in blue). We can further simplify the AC by pushing the $+$-nodes downwards (distributivity rule) and summing-out the leaf nodes (sum-out rule). We color-coded the removed $+$-nodes and leaf nodes in red in Figure 11. Finally, we apply the distributivity rule on $+(U_3)$-node on the right branch, which yields the simplified AC in Figure 7.

We can now apply the c-component method to derive projections for the AC nodes. In particular, The causal graph in Figure 3a induces the following c-factors:

$$Q[S_1] = \sum_{U_1,U_2,U_3} \theta_{U_1}\theta_{U_2}\theta_{U_3}\theta_{A|U_1,U_2}\theta_{B|U_2,U_3} \equiv \Pr(A,B)$$

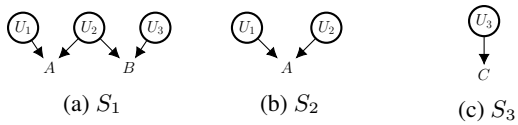
$$Q[S_2] = \sum_{U_1,U_2} \theta_{U_1}\theta_{U_2}\theta_{A|U_1,U_2} \equiv \Pr(A)$$

$$Q[S_3] = \sum_{U_3} \theta_{U_3}\theta_{C|U_3,X} \equiv \Pr(C|X)$$

Each of the above c-factors matches some node in the AC. Moreover, after plugging in the projections for the AC nodes, we find an invariant-cut as shown in Figure 7. We thus conclude that the causal effect is identifiable.

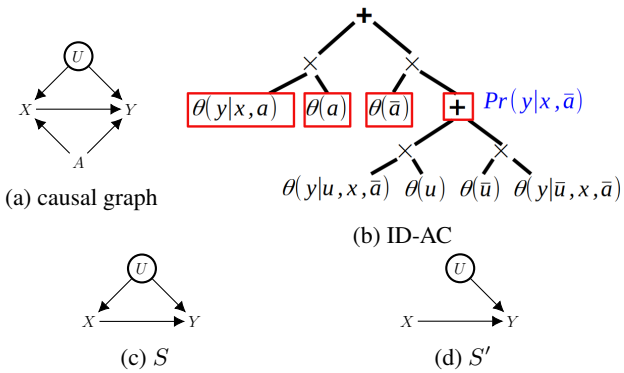
C.2 Additional Example on CSI

The following example is adapted from (Tikka, Hyttinen, and Karvanen 2019). Consider the causal graph G in Figure 13a with $\mathbf{V} = \{A, X, Y\}$. The causal effect $\Pr_x(y)$ is not identifiable under positivity constraint $\Pr(\mathbf{V}) > 0$.



(a) S_1 (b) S_2 (c) S_3

Figure 12: $\mathbf{V}$-computable subgraphs for Figure 3a.



(a) causal graph (b) ID-AC

(c) S (d) S'

Figure 13: CSI example adapted from (Tikka, Hyttinen, and Karvanen 2019) and an AC with an invariant-cut for $\Pr_x(y)$.

However, the causal effect becomes identifiable under the CSI constraints ($X \perp\!\!\!\perp U|\bar{a}$) and ($Y \perp\!\!\!\perp U|a, X$). Figure 13b depicts an AC with an invariant-cut for the causal effect.

We can find the projections in Figure 13b using the c-component method *but with a slight modification to leverage the CSI constraints*. In particular, consider the $+$-node on the right branch, which has the following expression:

$$\theta_u \theta_{y|u,x,\bar{a}} + \theta_{\bar{u}} \theta_{y|\bar{u},x,\bar{a}}$$

This expression does not directly match any c-factors induced by $\mathbf{V}$-computable subgraphs. However, the c-factor for the $\mathbf{V}$-computable subgraph S in Figure 13c

$$\sum_U \theta_U \theta_{Y|U,X,A} \theta_{X|U,A} \equiv \Pr(X, Y|A)$$

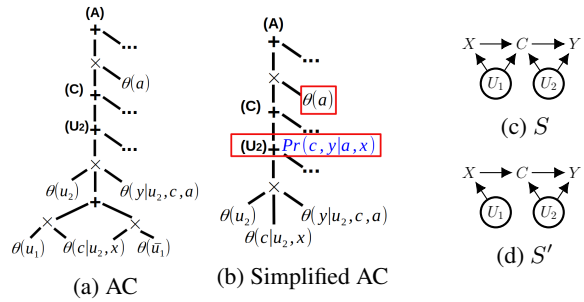
can be further simplified when we fix $A = \bar{a}$. In particular, the CSI constraint ($X \perp\!\!\!\perp U|\bar{a}$) allows us to replace $\theta_{X|\bar{a},U}$ with $\theta_{X|\bar{a}}$ in the c-factor, which yields

$$\sum_u \theta_u \theta_{Y|u,X,\bar{a}} \theta_{X|\bar{a}} \equiv \Pr(X, Y|\bar{a})$$

That is, the new graph S' in Figure 13d is also $\mathbf{V}$-computable when we fix $A = \bar{a}$. This allows us to further decompose S' into c-components by Definition 6 and attain the following c-factor and corresponding projection:

$$\sum_u \theta_u \theta_{Y|u,X,\bar{a}} \equiv \Pr(Y|X, \bar{a})$$

This example shows that CSI constraints may empower additional c-component decompositions and hence discover a larger set of $\mathbf{V}$-computable subgraphs.



(a) AC (b) Simplified AC (c) S (d) S'

Figure 14: (Simplified) AC constructed for $\Pr_x(y)$ wrt the causal graph in Figure 8b.

C.3 Derivation For Figure 8

We show that the causal effect $\Pr_x(y)$ is identifiable wrt the causal graph G in Figure 8b under the CSI constraint ($C \perp\!\!\!\perp U_1|x, U_2$). We first construct the AC for $\Pr_x(y)$ in Figure 14a, which incorporates the CSI constraint with the method in Section 6.1. We then simplify the AC using the simplification rules in Appendix A, which yields Figure 14b.

To find an invariant-cut, we need to find a projection for the following expression:

$$\sum_{u_2} \theta_{u_2} \theta_{c|u_2,x} \theta_{y|u_2,A,C}$$

which does not correspond to any c-factors directly. We now consider the c-factor of the $\mathbf{V}$-computable subgraph S in Figure 14c

$$\begin{aligned} Q[S] &= \sum_{U_1,U_2} \theta_{U_1} \theta_{U_2} \theta_{X|U_1,A} \theta_{C|U_1,U_2,X} \theta_{Y|U_2,A,C} \\ &\equiv \Pr(X, C, Y|A) \end{aligned}$$

If we fix $X = x$, we can leverage the CSI constraint and replace $\theta_{C|U_1,U_2,x}$ with $\theta_{C|U_2,x}$, which yields

$$\begin{aligned} &\sum_{u_1,u_2} \theta_{u_1} \theta_{u_2} \theta_{x|u_1,A} \theta_{C|u_1,u_2,x} \theta_{Y|u_2,A,C} \\ &= \sum_{u_1,u_2} \theta_{u_1} \theta_{u_2} \theta_{x|u_1,A} \theta_{C|u_2,x} \theta_{Y|u_2,A,C} \\ &\equiv \Pr(x, C, Y|A) \end{aligned}$$

Hence, the subgraph S' in Figure 13d is also $\mathbf{V}$-computable under $X = x$. We can further decompose S' into c-components and obtain the following c-factor:

$$\sum_{u_2} \theta_{u_2} \theta_{C|u_2,x} \theta_{Y|u_2,A,C} \equiv \Pr(C, Y|A, x)$$

This c-factor can now be used as a projection for the $+(U_2)$-node in the AC in Figure 8b. We can now find an invariant-cut (marked in red) and get an identifying formula $\Pr_x(y) = \sum_a \Pr(a) \sum_c \Pr(c, y|a, x)$.

C.4 Derivation For Figure 9b

We show that the AC in Figure 10 constructed for $\Pr_x(y)$ under $\Pr^\star(\mathbf{V})$ in Figure 9b can be simplified to an AC with a

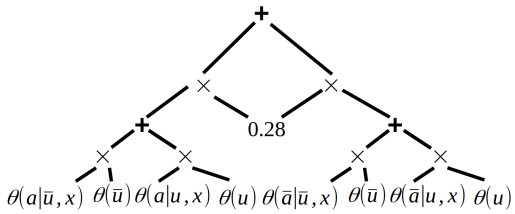


Figure 15: Simplified AC from Figure 10.

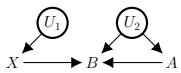


Figure 16: $\mathbf{V}$-computable subgraph for Figure 9c under $X = x, A = a$.

single constant. First, by evaluating the constants and merging the nodes, we obtain the AC in Figure 15. If we further push down the $+$-node at the root using the distributivity rule and apply the sum-out rule, we obtain an AC with a single constant 0.28.

C.5 Derivation For Figure 9c

Given the AC in Figure 10b, we still need to find the projection for the $+$-nodes with the following expression: $\sum_{u_2} \theta_{u_2}\theta_{\bar{a}|u_2}\theta_{B|u_2,\bar{a},x}$, which does not match any c-factors for $\mathbf{V}$-computable subgraphs. However, by exploiting the CSI relation, we can set $A = \bar{a}, X = x$ in the following c-factor $\sum_{U_1,U_2} \theta_{U_1}\theta_{U_2}\theta_{X|U_1}\theta_{A|U_2}\theta_{B|U_1,U_2,X,A}$ and get

$$\sum_{u_1,u_2} \theta_{u_1}\theta_{u_2}\theta_{x|u_1}\theta_{\bar{a}|u_2}\theta_{B|u_1,u_2,x,\bar{a}}$$
$$= \sum_{u_1,u_2} \theta_{u_1}\theta_{u_2}\theta_{x|u_1}\theta_{\bar{a}|u_2}\theta_{B|u_2,x,\bar{a}} \equiv \Pr(x, \bar{a}, B)$$

That is, once we fixed the values of X and A, the c-component S in Figure 16. We can now further decompose S into two c-component, one of which is exactly the expression for the $+$-node in the AC:

$$\sum_{u_2} \theta_{u_2}\theta_{\bar{a}|u_2}\theta_{B|u_2,x,\bar{a}} \equiv \Pr(\bar{a}, B|x)$$

Hence, we conclude the projection fro the $+$-node is $\Pr(\bar{a}, b|x)$.

D Proofs

Proof of Proposition 1

Proof. The causal effect is unidentifiable under some $\Pr^\star(\mathbf{V}) > 0$ since the ID algorithm returns "unidentifiable" (i.e., the graph contains a *hedge*[17]). We next show that the causal effect $\Pr_x(y)$ is identifiable whenever $\Pr^\star(\mathbf{V})$ satisfies the CSI constraints $(Y \perp\!\!\!\perp C|a, B)$ and $(Y \perp\!\!\!\perp B|\bar{a}, C)$. We show that the $\Pr_x(y)$ can be identified as

$$\Pr_x(y) = \Pr(a, y) + \sum_c \Pr(y|\bar{a}, c)\Pr(c|x)\Pr(\bar{a})$$

[17]Hedge is a complete graphical structure that induces unidentifiability; see (Shpitser and Pearl 2006) for details.

We break $\Pr_x(y)$ into two parts $\Pr_x(a, y)$ and $\Pr_x(\bar{a}, y)$ by the law of total probability and identify each part separately. We first consider $\Pr_x(a, y)$.

$$\sum_b \sum_c \Pr(y|a, b, c) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)$$
$$\Pr(a|u_1, u_2) \sum_{u_3} \Pr(u_3)\Pr(c|u_3, x)\Pr(b|u_2, u_3)$$
$$= \sum_b \Pr(y|a, b) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)\Pr(a|u_1, u_2)$$
$$\sum_{u_3} \Pr(u_3)\Pr(b|u_2, u_3) \qquad (Y \perp\!\!\!\perp C|a, B)$$
$$= \sum_b \Pr(y|a, b) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)\Pr(a|u_1, u_2)\Pr(b|u_2)$$
$$(U_2 \perp\!\!\!\perp U_3)$$
$$= \sum_b \Pr(y|a, b) \sum_{u_2} \Pr(u_2)\Pr(a|u_2)\Pr(b|u_2)$$
$$(U_1 \perp\!\!\!\perp U_2)$$
$$= \sum_b \Pr(y|a, b)\Pr(a, b) \qquad (A \perp\!\!\!\perp B \mid U_2)$$
$$= \Pr(a, y)$$

We next consider $\Pr_x(\bar{a}, y)$.

$$\sum_b \sum_c \Pr(y|\bar{a}, b, c) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)\Pr(\bar{a}|u_1, u_2)$$
$$\sum_{u_3} \Pr(u_3)\Pr(c|u_3, x)\Pr(b|u_2, u_3)$$
$$= \sum_c \Pr(y|\bar{a}, c) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)\Pr(\bar{a}|u_1, u_2)$$
$$\sum_{u_3} \Pr(u_3)\Pr(c|u_3, x) \qquad (Y \perp\!\!\!\perp B \mid A = \bar{a}, C)$$
$$= \sum_c \Pr(y|\bar{a}, c) \sum_{u_1} \sum_{u_2} \Pr(u_1)\Pr(u_2)\Pr(\bar{a}|u_1, u_2)\Pr(c|x)$$
$$(U_3 \perp\!\!\!\perp X)$$
$$= \sum_c \Pr(y|\bar{a}, c)\Pr(\bar{a})\Pr(c|x) \qquad (U_1 \perp\!\!\!\perp U_2)$$

Hence, $\Pr_x(y)$ is identifiable when the CSI constraints hold. □

Proof of Proposition 2

WLG, to show that the causal effect $\Pr_x(y)$ is unidentifiable under any $\Pr^\star(X, Y) > 0$, we construct two parameterizations Θ^1 and Θ^2 that agree on any $\Pr^\star(X, Y)$ but disagrees on $\Pr_{x_1}(y_1)$, where $\{x_1, \ldots, x_n\}$ are the states of X and $\{y_1, \ldots, y_m\}$ are the states of Y. Moreover, we assume that the hidden variables U is binary and has two states u_1, u_2. We start by constructing a parameterization for G that satisfy certain conditions.

Lemma 1. *There exists a parameterization Θ for G that induces $\Pr^\star(X, Y)$ and in which (i) all CPTs are strictly positive; and (ii) $\theta_{x|u_1} \neq \theta_{x|u_2}$ for all states x of X.*

Proof. We first construct a parameterization Θ for the subgraph $X \to Y$ that induces exactly $\Pr^\star(X, Y)$. In particular, let $\theta_x = \Pr^\star(x)$ and $\theta_{y|x} = \Pr^\star(y|x)$ for all states x and y. Note that all values in Θ are in $(0, 1)$ since $\Pr^\star(X, Y) > 0$.

We next construct a parameterization Θ' for the subgraph $U \to X \to Y$ based on Θ. In particular, let $\epsilon \to 0^+$ be an arbitrarily small constant, we assign

$$\begin{aligned}
\theta'_{u_1} &= \theta'_{u_2} = 0.5 \\
\theta'_{x|u_1} &= \theta_x + \epsilon \\
\theta'_{x|u_2} &= \theta_x - \epsilon \\
\theta'_{y|x} &= \theta_{y|x}
\end{aligned}$$

for all states x, y. Let $\Pr'$ be the distribution induced by Θ', we show that $\Pr' = \Pr^\star$ as follows.

$$\begin{aligned}
\Pr'(x, y) &= \theta'_{u_1}\theta'_{x|u_1}\theta'_{y|x} + \theta'_{u_2}\theta'_{x|u_2}\theta'_{y|x} \\
&= 0.5(\theta_x + \epsilon)\theta_{y|x} + 0.5(\theta_x - \epsilon)\theta_{y|x} \\
&= \theta_x\theta_{y|x} = \Pr^\star(x, y).
\end{aligned}$$

By construction, Θ' satisfies both conditions (i) & (ii).

We finally construct Θ'' based on Θ' by assigning the same CPTs for U and X but different CPTs for Y. In particular,

$$\theta''_{y|u_1,x} = \theta''_{y|u_2,x} = \theta'_{y|x}$$

Θ'' also induces the distribution $\Pr^\star$ since $(Y \perp\!\!\!\perp U|X)$. □

Proof of Proposition 2. Let Θ^1 be parameterization that satisfies the conditions in Lemma 1, we next construct Θ^2 that also induces $\Pr^\star(X, Y)$ but disagrees with Θ^1 on the causal effect $\Pr_{x_1}(y_1)$. We do so by assigning same CPTs for U and X, i.e., $\Theta^2_U = \Theta^1_U$ and $\Theta^2_X = \Theta^1_X$, but different CPTs for Y. Specifically, for each state $y \neq y_1$, we assign

$$\theta^2_{y|u_1,x} = \begin{cases} \theta^1_{y|u_1,x} + \epsilon & \text{if } y = y_1 \\ \theta^2_{y|u_1,x} = \theta^1_{y|u_1,x} - \frac{\epsilon}{m-1} & \text{if } y \neq y_1 \end{cases}$$

and

$$\theta^2_{y|u_2,x} = \begin{cases} \theta^1_{y|u_2,x} - \frac{\Pr^1(u_1,x_1)}{\Pr^1(u_2,x_1)} & \text{if } y = y_1 \\ \theta^1_{y|u_2,x} + \frac{\Pr^1(u_1,x_1)}{\Pr^1(u_2,x_1)}\frac{\epsilon}{(m-1)} & \text{if } y \neq y_1 \end{cases}$$

where $\Pr^1$ denotes the distribution induced by Θ^1.

We first prove show that Θ^2 induces a distribution $\Pr^2$ where $\Pr^2(X, Y) = \Pr^1(X, Y) = \Pr^\star(X, Y)$. We first consider the instantiations that contains y_1,

$$\begin{aligned}
&\Pr^2(x, y_1) = \theta^2_{u_1}\theta^2_{x|u_1}\theta^2_{y_1|u_1,x} + \theta^2_{u_2}\theta^2_{x|u_2}\theta^2_{y_1|u_2,x} \\
&= \theta^1_{u_1}\theta^1_{x|u_1}(\theta^1_{y_1|u_1,x} + \epsilon) + \theta^1_{u_2}\theta^1_{x|u_2}(\theta^1_{y_1|u_2,x} - \frac{\Pr^1(u_1, x)}{\Pr^1(u_2, x)}\epsilon) \\
&= \Pr^1(x, y_1).
\end{aligned}$$

The last step is due to the fact that $\theta^1_{u_1}\theta^1_{x|u_1} = \Pr(u_1, x)$.

We next consider the case when $y \neq y_1$.

$$\begin{aligned}
&\Pr^2(x, y) = \theta^2_{u_1}\theta^2_{x|u_1}\theta^2_{y|u_1,x} + \theta^2_{u_2}\theta^2_{x|u_2}\theta^2_{y|u_2,x} \\
&= \theta^1_{u_1}\theta^1_{x|u_1}(\theta^1_{y|u_1,x} - \frac{\epsilon}{m-1}) + \\
&\theta^1_{u_2}\theta^1_{x|u_2}(\theta^1_{y|u_2,x} + \frac{\Pr^1(u_1, x)}{\Pr^1(u_2, x)}\frac{\epsilon}{(m-1)}) \\
&= \Pr^1(x, y).
\end{aligned}$$

Together with the previous result, we have $\Pr^2(X, Y) = \Pr^\star(X, Y)$. We are left to show that Θ^1 and Θ^2 induce different causal effect, i.e., $\Pr^1_{x_1}(y_1) \neq \Pr^2_{x_1}(y_1)$, which can be proved as follows.

$$\begin{aligned}
&\Pr^2_{x_1}(y_1) = \theta^2_{u_1}\theta^2_{y_1|u_1,x_1} + \theta^2_{u_2}\theta^2_{y_1|u_2,x_1} \\
&= \theta^1_{u_1}(\theta^1_{y_1|u_1,x_1} + \epsilon) + \theta^1_{u_2}(\theta^1_{y_1|u_2,x_1} - \frac{\Pr^1(u_1, x_1)}{\Pr^1(u_2, x_1)}\epsilon) \\
&= \Pr^1_{x_1}(y_1) + \theta^1_{u_1}\epsilon - \theta^1_{u_2}\frac{\Pr^1(u_1, x_1)}{\Pr^1(u_2, x_1)}\epsilon \\
&= \Pr^1_{x_1}(y_1) + \frac{\Pr^1(u_1)(\Pr^1(x_1|u_2) - \Pr^1(x_1|u_1))}{\Pr^1(x_1|u_2)}\epsilon \\
&\neq \Pr^1_{x_1}(y_1).
\end{aligned}$$

since $\Pr^1$ satisfies Conditions (i) & (ii) in Lemma 1. □

Proof of Proposition 3

The result follows from Definition 3 and Definition 4.

Proof of Proposition 4

Proof. By Proposition 3, it suffices to show that the output of the AC is $\langle G, \mathbf{V}, \mathcal{M}\rangle$-invariant. If we replace all the expressions of nodes on the invariant-cut with their projections, it is guaranteed that the output expression of the AC (with projections) only involves variables $\mathbf{V}$, which is determined by $\Pr(\mathbf{V})$. □

Proof of Theorem 1

Proof. First note that the completeness proof of the ID algorithm (Shpitser and Pearl 2006; Huang and Valtorta 2006) assumes all variables are binary. Hence, the if direction of the theorem must hold.

Before proving the only-if direction, we briefly review the complete causal-effect identifiability algorithm in (Huang and Valtorta 2006). The algorithm starts by omitting all variables that are not ancestors of $\mathbf{Y}$ in the mutilated graph $G_{\overline{\mathbf{X}}}$ and decomposing the mutilated graph into into c-components $\mathcal{C} = \{C_1, \ldots, C_k\}$.[18] The goal is then to identify the c-factor for each of the c-component in $\mathcal{C}$. This is implemented through a recursive procedure that identifies some $C \in \mathcal{C}$ from another graph T. In each recursive call, two operations may be performed: (i) Assign $T = An(C)_T \cap T$ where $An(C)_T$ denotes the variables in T that

[18] $G_{\overline{\mathbf{X}}}$ is obtained from the original causal graph G by removing the incoming edges of $\mathbf{X}$.

are ancestors of variables in C; and (ii) decompose T into c-components and identify C from one of the c-components. We say that C is identifiable from T iff $T = C$ when the recursion terminates. The causal effect is identifiable iff all c-components in $\mathcal{C}$ can be identified from G.

One key observation is that both operations are captured by the cases in Definition 6. Operation (i) is equivalent to recursively pruning leaf nodes that are not in C, and operation (ii) is the same as case (3) in Definition 6. Hence, a c-component $C \in \mathcal{C}$ is identifiable only if C is a $\mathbf{V}$-computable subgraph of G, and we can always find a projection for C if it can be identified by the complete algorithm in (Huang and Valtorta 2006).

We now construct an AC based on the expression $\sum_{\mathbf{V}\setminus(\mathbf{X}\cup\mathbf{Y})} \prod_{C\in\mathcal{C}} Q[C]$, where $Q[C]$ denotes the c-factor for the c-component C and can be represented as an AC in a standard way. Note that the AC always computes the causal effect under all models. Moreover, if the causal effect is identifiable, we can always find projections for the AC nodes corresponding to these $Q[C]$'s, which form an invariant-cut for the AC. $\square$