Divide and Conquer for Causal Computation

Bjøru, A. R., Cabañas, R., Langseth, H., & Salmerón, A.

Published in:
International Journal of Approximate Reasoning

DOI (link to publication from Publisher):
https://doi.org/10.1016/j.ijar.2025.109520

Publication date:
2025

Document Version:
Accepted author manuscript, peer reviewed version

Divide and Conquer for Causal Computation

Anna Rodum Bjøru[a,*], Rafael Cabañas[b], Helge Langseth[a], Antonio Salmerón[b]

[a]*Department of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway*
[b]*Department of Mathematics and Centre for the Development and Transfer of Mathematical Research to Industry (CDTIME), University of Almería, Almería, Spain*

Abstract

Structural causal models are a powerful framework for causal and counterfactual inference, extending the capabilities of traditional Bayesian networks. These models comprise endogenous and exogenous variables, where the exogenous variables frequently lack clear semantic interpretation. Exogenous variables are typically unobservable, rendering certain counterfactual queries unidentifiable. In such cases, standard inference algorithms for Bayesian networks are insufficient. Recent methods attempt to bound unidentifiable queries through imprecise estimation of exogenous probabilities. However, these methods become computationally infeasible as the cardinality of the exogenous variables increases, thereby constraining the complexity of applicable models. In this paper we study a divide-and-conquer approach that decomposes a general causal model into a set of submodels with low-cardinality exogenous variables, enabling exact calculation of any query within these submodels. By aggregating results from the submodels, efficient approximations

*Corresponding author: anna.r.bjoru@ntnu.no

of bounds for queries in the original model are obtained. Our proposal is able to handle models with variables of any cardinality assuming that there are no unobserved confounders. We show that the method is theoretically robust, and experimental results demonstrate that it achieves more accurate bounds with lower computational costs compared to existing techniques.
Keywords: Structural causal models; causality; counterfactual reasoning; Satisfiability; Heuristic search.

---

1. Introduction

Structural causal models (SCMs) with discrete variables [1, 2] are a type of probabilistic graphical models (PGM) for causal and counterfactual reasoning. SCMs enable reasoning about hypothetical scenarios, such as estimating the probability of recovery for a deceased patient in a medical trial if they had received a different treatment. SCMs consist of endogenous (observable) and exogenous (usually latent) variables, with endogenous values determined from exogenous ones through structural equations. Often, the exogenous probabilities are unavailable due to the lack of data for these variables. Consequently, many queries are non-identifiable and cannot be calculated.

One of the first approaches for addressing this problem presented a systematic technique to derive constraints on a causal query, albeit with exponential growth [3]. Later on, methods for deriving bounds on causal effects [4] and for approximating credible intervals based on sampling algorithms [5] where proposed. More closely related to our work is the idea of transforming SCMs into credal networks [6], requiring the solution of various linear

programming problems [7]. However, this approach may be infeasible due to the large cardinality of exogenous variables. More recently, EMCC was introduced for approximating the bounds of any non-identifiable query [8]. This involves repeatedly running the expectation-maximization (EM) algorithm [9] to obtain precise specifications of exogenous distributions. Queries can then be separately calculated and aggregated to approximate the bounds. The problem with EMCC is that each individual EM run necessitates an exceptionally large number of iterations to achieve low error.

Recently [10] we introduced the *Divide and Conquer for Causal Computation* (DCCC) method, which integrates elements of the two previously mentioned approaches. It aims to obtain precise specifications of exogenous distributions, from which any query can be calculated. DCCC reduces SCMs by removing certain exogenous states, transforming the SCMs into collections of less complex models. Then, various linear programming problems with unique solutions are solved in the reduced models. This paper is an extended version of the conference paper [10]. We now introduce the methodological results that guarantee the theoretical soundness of DCCC, generalize the method to non-binary variables and carry out additional experiments.

The remainder of the paper is organized as follows. Section 2 contains the fundamental definitions and notation related to SCMs and counterfactual reasoning. The characterization of SCMs as Bayesian networks where the conditional distributions contain imprecise probabilities is described in Section 3. The theoretical contributions of this paper are contained in Section 4, while Section 5 is devoted to the experimental evaluation of our proposal. The paper ends with conclusions and future research lines in Section 6.

2. Background

2.1. Basic notation

With respect to the general notation, upper-case letters are used to denote random variables and lower-case for their possible values (or states). That is, given a variable V, v is an element of its domain, denoted by Ω_V. We assume that all the variables are discrete. Similarly, $\mathbf{V} = \{V_1, V_2, \ldots, V_n\}$ denotes a set of variables and $\mathbf{v}$ a joint state of its domain $\Omega_{\mathbf{V}} = \times_{V \in \mathbf{V}} \Omega_V$. For simplicity, variables are omitted from assignments when their context is clear. For instance, $P(V = v)$ can sometimes be denoted simply as $P(v)$. In a directed graph, Pa_V are the parents (i.e., the immediate predecessors) of V. The notation $\tilde{P}$ is used to denote empirical distributions.

2.2. Structural causal models

Structural Causal Models (SCMs) [1] are a class of probabilistic graphical models (PGMs) used for causal and counterfactual reasoning, consisting of two types of nodes: *endogenous* nodes, which represent the internal variables of the modeled problem, and *exogenous* nodes, which represent factors outside the model. SCMs can be more formally defined as follows [2].

Definition 1 (Structural causal model (SCM)). A structural causal model $\mathcal{M}$ is a 4-tuple $\langle \mathbf{U}, \mathbf{V}, \mathcal{F}, \mathcal{P} \rangle$, where

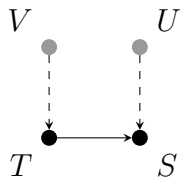
- $\mathbf{U}$ is a set of exogenous variables that are determined by factors outside the model,
- $\mathbf{V}$ is a set of endogenous variables $\{V_1, V_2, \ldots, V_n\}$ that are determined by other (exogenous and endogenous) variables in the model, i.e. by variables in $\mathbf{U} \cup \mathbf{V}$,

- $\mathcal{F}$ is a set of functions $\{f_{V_1}, f_{V_2}, ..., f_{V_n}\}$ called *structural equations* (SE), such that each of them is a function $f_{V_i} : \Omega_{\mathbf{U}_i} \cup \Omega_{\mathrm{Pa}_{V_i}} \rightarrow \Omega_{V_i}$, where $\mathrm{Pa}_{V_i} \subseteq \mathbf{V}$ are the endogenous variables directly determining V_i and $\mathbf{U}_i \subseteq \mathbf{U}$ are the exogenous variables directly determining V_i,
- $\mathcal{P}$ is a set of probability distributions $P(U)$ for each $U \in \mathbf{U}$.

The structural equations $\mathcal{F}$ define a directed acyclic graph (DAG) $\mathcal{G}$, referred to as the *causal graph* of the model, where the nodes correspond to the variables in $\mathbf{U} \cup \mathbf{V}$. The domains of both the endogenous and exogenous variables are assumed to be finite and discrete. A discrete exogenous variable, taking values on a finite set of unobserved states, is enough to represent all counterfactual distributions [5].

When the distributions for the exogenous variables are not known, we refer to the model as a partially specified SCM, denoted by calligraphic letters, such as $\mathcal{M}$. In contrast, if all such distributions are provided, the model is considered fully specified and is denoted as, e.g., M. If it is not explicitly stated whether a model is fully specified or not, the most general representation is used, namely $\mathcal{M}$. As an example, consider the SCM shown in Figure 1, which models a drug study involving 700 patients [11]. The causal graph on the left includes the endogenous variables $\mathbf{V} = \{T, S\}$, representing the *treatment* and *survival*, respectively. The goal is to analyze whether receiving treatment ($T = 1$) contributes to survival ($S = 1$). Meanwhile, $\mathbf{U} = \{V, U\}$ represents the set of exogenous variables, which are assumed to be root nodes, with endogenous variables as their children.

The SCM depicted in Figure 1 assumes *unconfoundedness*, meaning that there are no unobserved confounders, i.e., no unobserved common causes of



f_T

T	v_0	v_1
0	1	0
1	0	1

$\tilde{P}(T)$	
$T=0$	0.337
$T=1$	0.663

f_S

T	S	u_0	u_1	u_2	u_3
0	0	1	1	0	0
0	1	0	0	1	1
1	0	1	0	1	0
1	1	0	1	0	1

$\tilde{P}(S\|T)$	$T=0$	$T=1$
$S=0$	0.462	0.323
$S=1$	0.538	0.677

Figure 1: Elements of an SCM: (left) causal graph, (center) structural equations and (right) empirical distribution computed from the data.

the treatment and the outcome. Such an SCM is termed *Markovian* [12]. Graphically, this indicates that each exogenous variable has only one endogenous child. In this paper, only Markovian models will be considered.

Figure 1 (center) shows the SEs $f_T(V)$ and $f_S(T,U)$ as deterministic CPTs of the form $\tilde{P}(T|V)$ and $\tilde{P}(S|T,U)$, characterized by containing only ones and zeros. If not provided from expert knowledge, SEs can be automatically inferred from the causal graph, without any loss of generality, via a *canonical specification*. This is the case of the SEs shown in the example, where the states of an exogenous variable will then represent all possible deterministic mechanisms between its children and their respective endogenous parents. Conversely, under the non-canonical specification, some of the exogenous states are assumed to be impossible and directly omitted from the CPTs. In a Markovian model, the cardinality of each exogenous variable U is at most $|\Omega_Y|^{|\Omega_{\mathbf{X}}|}$, where Y is its only child and $\mathbf{X}$ the set of endogenous parents of Y. For the canonical models we are interested in this paper, we

have that $|\Omega_U| = |\Omega_Y|^{|\Omega_{\mathbf{X}}|}$.

When doing inference with an SCM $\mathcal{M}$ and a dataset $\mathcal{D}$, typically only observations for the endogenous variables are available. For example, in the model under consideration, it is possible to calculate from $\mathcal{D}$ the empirical distributions $\tilde{P}(T)$ and $\tilde{P}(S|T)$ with the values shown in Figure 1 (right). The task of doing inference in a partial SCM given a dataset involves estimating a set of fully-specified SCMs that are compatible with the data. This idea, which is fundamental for our method, is extended in the following section.

2.3. Counterfactual reasoning

Typically, performing probabilistic inference in PGMs involves computing the posterior probability distribution for a set of variables of interest given evidence about some other variables, which is called the *observational query*. Causal reasoning usually involves handling hypothetical scenarios in which the probability of a variable given that we intervene on another can be computed, which is called an *interventional query*. This is achieved using the so-called *do calculus* [1], which requires us to introduce the intervention operator, denoted as $do(\cdot)$, which means that some variable is forced to take one specific value.

In a more general setup, *counterfactual* inference allows handling scenarios in which the same variable can be both observed and subject to intervention. In what follows, if x is a possible value of a variable X, we will denote by x' the complement of x. We will denote by $Y_x = y$ the event consisting in variable Y taking value y in a scenario where X is intervened to take the value x. A *counterfactual query* $P(Y_x = y|X = x', Y = y')$ represents the

probability of event y occurring in a hypothetical scenario where the variable X is intervened to take the value x, given that in reality, events x' and y' occurred. The notation $Y_{X=x}$ or simply Y_x denotes the variable Y in the hypothetical scenario, whereas variables in the reality remain with the name unchanged.

Typical counterfactual queries are related to the abstract concepts of *necessity* (when something must occur in order for something else to happen) and *sufficiency* (when something is enough for something else to take place). Note that, in the following definitions, we are interpreting $X = x$ (resp. $Y = y$) as X occurring (resp. Y occurring), while $X = x'$ (resp. $Y = y'$) means that X does not occur (resp. that Y does not occur).

Definition 2 (Probability of necessity [1]). Let X and Y be two variables in an SCM. The *probability of necessity* is defined as

$$\mathrm{PN}(X, Y) = P(Y_{x'} = y' | X = x, Y = y). \tag{1}$$

PN can be interpreted as the probability that Y would not have taken value y if X had not taken value x, given that in fact variables X and Y took values x and y respectively. In other words, it measures to what extent it is necessary that X takes value x for Y to take value y.

The abstract concept of sufficiency is addressed by the following definition. It measures the capacity of X taking value x causing Y to take value y.

Definition 3 (Probability of sufficiency [1]). Let X and Y be two variables in an SCM. The *probability of sufficiency* is defined as

$$\mathrm{PS}(X, Y) = P(Y_x = y | X = x', Y = y'). \tag{2}$$

The following definition is closely related to PN and PS. It measures how Y reacts to X, hence expressing to what extent $X = x$ is necessary and sufficient for $Y = y$.

Definition 4 (Probability of necessity and sufficiency [1]). Let X and Y be two variables in an SCM. The *probability of necessity and sufficiency* is defined as

$$\text{PNS}(X, Y) = P(Y_x = y, Y_{x'} = y'). \tag{3}$$

The counterfactual queries defined above are *unidentifiable* when only observational data is available [13], which means that their exact value cannot be computed.

3. Imprecise characterisation

It has been shown [7, 8] that an SCM can be accurately mapped into a *credal network*, which essentially is a generalization of a Bayesian network with an imprecise specification of its parameters [6]. In this context, each node is associated with a set of probability mass functions known as a *credal set*. Specifically, when mapping SCMs into credal networks, the endogenous observations impose linear constraints on the probabilities of the exogenous variables. Consequently, we can derive a separate credal set for each exogenous variable, which here will be called the *solution set*, formally defined as follows.

Definition 5 (Solution set for an exogenous variable). A solution for an exogenous variable U and a dataset $\mathcal{D}$ is the credal set defined as

$$\mathcal{K}(U) := \left\{ P(U) : \sum_{u \in \Omega_U} P(u) \cdot P(Y|\mathbf{X}, u) = \tilde{P}(Y|\mathbf{X}) \right\} \tag{4}$$

where Y is the only child of U, $\mathbf{X}$ the set of endogenous parents of Y and $\tilde{P}(Y|\mathbf{X})$ is the empirical distribution computed from $\mathcal{D}$.

In other words, $\mathcal{K}(U)$ is the convex set of all the distributions over U leading to the same distribution over the endogenous children after marginalizing out the exogenous parents. In our setting, we are interested in finding all these distributions. Each element $P(U) \in \mathcal{K}(U)$ will be called a solution for U. This idea can be extended to an SCM as follows:

Definition 6 (Solution for an SCM). A solution for a partially-specified SCM $\mathcal{M}$ with exogenous variables $\mathbf{U}$ given dataset $\mathcal{D}$ is a set of distributions $\{P(U)\}_{U_\in \mathbf{U}}$ where each $P(U)$ is in $\mathcal{K}(U)$, i.e., it is a solution for the given exogenous variable.

Example 1. *For the model $\mathcal{M}$ and the empirical distribution (from a dataset $\mathcal{D}$) shown in Figure 1, there is a unique solution for variable V, namely $\mathcal{K}(V) = \{P(V) = [0.337, 0.663]\}$. On the other hand, solutions for U are all the distributions that are convex combinations of $P_1(U)$ and $P_2(U)$, defined as*

$$P_1(U) = \begin{array}{cccc} u_0 & u_1 & u_2 & u_3 \\ \Big[0.323 & 0.139 & 0 & 0.538\Big] \end{array}, \quad P_2(U) = \begin{array}{cccc} u_0 & u_1 & u_2 & u_3 \\ \Big[0 & 0.462 & 0.323 & 0.215\Big] \end{array}.$$

solving the set of equations

$$P(u_0) + P(u_1) = 0.462$$
$$P(u_2) + P(u_3) = 0.538$$
$$P(u_0) + P(u_2) = 0.323$$
$$P(u_1) + P(u_3) = 0.677$$

Intuitively, each fully-specified SCM that is a solution is also a model that could have produced the available endogenous data. Thus, we can introduce the following solvability condition:

Definition 7 (Solvable SCM). A partially-specified SCM is *solvable* for a given dataset $\mathcal{D}$ iff there exists a non-empty solution set $\mathcal{K}(U)$ for each exogenous variable $U \in \mathbf{U}$.

In other words, a partially-specified SCM is solvable (for the available endogenous data) if there exists at least one distribution for each exogenous variable satisfying the linear constraints. When this happens, it is said that the dataset is *M-compatible* [8]. Frequently, the solution for an SCM will not be unique, and hence it is required to define a solution set for an SCM as follows.

Definition 8 (Solution set for an SCM). A solution set for a partially-defined SCM $\mathcal{M}$ given dataset $\mathcal{D}$, denoted $\mathcal{S}_{\mathcal{M},\mathcal{D}}$, is the set of all the fully-specified SCMs such that $P(U) \in \mathcal{K}(U)$ for each $U \in \mathbf{U}$.

That is, each SCM contained in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ is a solution for the given dataset and partially-specified SCM. This set can be represented as a credal network where each exogenous variable has associated a credal set as defined by Equation (4) whereas the endogenous variables, instead of a credal set, have associated a single conditional distribution corresponding to the SEs. Intuitively, this particular model represents all the fully-specified SCMs that might have produced the available endogenous data. Typical counterfactual queries in an SCM are *probability of necessity* (PN, Definition 2) and *the*

probability of sufficiency (PS, Definition 3). For further details see [1, 14]. Any counterfactual query $q_{\mathcal{M},\mathcal{D}}$ can be transformed into a query in the credal network defined as

$$q_{\mathcal{M},\mathcal{D}} := \{q_M \mid M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}\}, \tag{5}$$

where q_M is the same query, but computed in each fully-specified SCM M that is a member of the solution set. In practice, the previous set of queries will be summarized by the lower and upper bounds:

$$\left[\min_{M\in\mathcal{S}_{\mathcal{M},\mathcal{D}}} q_M, \max_{M\in\mathcal{S}_{\mathcal{M},\mathcal{D}}} q_M\right]. \tag{6}$$

Example 2. *In the context of the running example, the set of models contained in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ is any fully-specified SCM where $P(v_0) = 0.337, P(v_1) = 0.663$ and $P(U)$ is a convex combination of $P_1(U)$ and $P_2(U)$. Then PS is bounded to the interval* $[0.301, 1.0]$.

4. The divide and conquer algorithm

4.1. Model reduction

The underlying idea of our method consists of transforming a complex SCM into a simpler one with exogenous variables of smaller cardinality. The transformation proposed is essentially the removal of some states from exogenous domains, namely a *reduction*[1], which can be defined as follows.

Definition 9 (Reduction operator). Let $\mathcal{M}$ be a partially-specified SCM whose set of exogenous variables is $\mathbf{U}$ and let $u \in \Omega_U$ with $U \in \mathbf{U}$. Then the

[1]What we call *reduction* in the following is sometimes referred to as *branching* in the optimization literature.

reduction operation, denoted $R(\mathcal{M}, u)$, produces a new partially-specified SCM $\mathcal{M}'$ by removing assignments from $\mathcal{F}$ and $\mathcal{P}$ in $\mathcal{M}$ for which $U = u$.

This reduction operation will be applied to various states of the different exogenous variables, i.e., to a set defined as $\mathcal{A}_{\mathbf{U}} := \left\{u^{(i)}\right\}_{i=1}^{m}$ s.t. $u^{(i)} \in \Omega_U$ and $U \in \mathbf{U}$. For simplicity we can recursively define this as $R(\mathcal{M}, \mathcal{A}_{\mathbf{U}}) = R\left(R(\mathcal{M}, u^{(1)}), \mathcal{A}_{\mathbf{U}} \setminus \{u^{(1)}\}\right)$. The reduction operation $R(\mathcal{M}, u)$ is equivalent to imposing the constraint $P(u) = 0$ on $\mathcal{M}$. Thus, when looking for the solution set of a reduced SCM, we can equivalently look for all the solutions from the original solution set that are consistent with that constraint. The following theorem simplifies calculating the set of queries given in Equation (5).

Theorem 1. *Let $\mathcal{M}$ be a partially-specified SCM, and $\mathcal{D}$ an M-compatible dataset. Let $\mathcal{A}_{\mathbf{U}}$ be a set of possible values of some variables in $\mathbf{U}$. If $\mathcal{M}' = R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ is solvable for $\mathcal{D}$, then it holds that $\mathcal{S}_{\mathcal{M}',\mathcal{D}} \subseteq \mathcal{S}_{\mathcal{M},\mathcal{D}}$.*

Proof. Any fully-specified model contained in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ by definition satisfies the linear constraints specified in Equation (4) for each $U \in \mathbf{U}$. The same constraints are satisfied by any member of $\mathcal{S}_{\mathcal{M}',\mathcal{D}}$. Additionally, any fully-specified SCM in the latter set also respects the additional constraints imposed by $R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ stating that $P\left(u^{(i)}\right) = 0$ on $\mathcal{M}$ for all $\mathcal{A}_{\mathbf{U}} = \{u^{(i)}\}_{i=1}^{m}$ s.t $u^{(i)} \in \Omega_U$ and $U \in \mathbf{U}$. Given that the set of constraints associated to $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ is a subset of those associated to $\mathcal{S}_{\mathcal{M}',\mathcal{D}}$, it holds that if $M \in \mathcal{S}_{\mathcal{M}',\mathcal{D}}$ then $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$, and the result follows. □

Corollary 1. *Let $\mathcal{M}$ be a partially-specified SCM, and $\mathcal{D}$ an M-compatible dataset. If $\mathcal{M}' = R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ is solvable for $\mathcal{D}$, then it holds that $q_{\mathcal{M}',\mathcal{D}} \subseteq q_{\mathcal{M},\mathcal{D}}$.*

Proof. According to Equation (5), and taking into account that, as stated by Theorem 1, $\mathcal{S}_{\mathcal{M}',\mathcal{D}} \subseteq \mathcal{S}_{\mathcal{M},\mathcal{D}}$,

$$q_{\mathcal{M}',\mathcal{D}} = \{q_M \mid M \in \mathcal{S}_{\mathcal{M}',\mathcal{D}}\} \subseteq \{q_M \mid M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}\} = q_{\mathcal{M},\mathcal{D}}.$$

□

Example 3. *If the reduction $R(\mathcal{M}, u_2)$ is applied to the running example, the solution set for U is $\mathcal{K}(U) = \{[P(u_0) = 0.323, P(u_1) = 0.139, P(u_3) = 0.538]\}$ and PS is 0.301, which is the lower bound in Example 2.*

In practice, it is not necessary to compute all the elements in $\mathcal{S}_{\mathcal{M}',\mathcal{D}}$. We will only consider the *extreme models*. These are fully-specified SCMs where each $P(U)$ is an extreme point of $\mathcal{K}(U)$. Formally, we define an extreme model in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ as follows:

Definition 10 (Extreme models). Let $M_0 \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$. We say that M_0 is an *internal model* in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ if there exist two fully-specified SCMs $M_1, M_2 \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$ and a real number $\lambda \in (0, 1)$ s.t.

$$P_0(\mathbf{u}) = \lambda P_1(\mathbf{u}) + (1 - \lambda)P_2(\mathbf{u}), \qquad \forall \mathbf{u} \in \Omega_{\mathbf{U}},$$

where we use $P_i(\mathbf{U})$ to denote the probability distribution allocated to the exogenous variables by model M_i, $i = 0, 1, 2$. Finally, M_0 is *extreme* in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ if and only if it is not internal in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$.

The following result shows that it is enough to explore the extreme models when computing bounds for any probability $P(\mathbf{u}), \mathbf{u} \in \Omega_{\mathbf{U}}$.

Theorem 2. *Let $\mathcal{M}$ be a partially-specified SCM and $\mathcal{D}$ be an M-compatible dataset. It holds that for any $\mathbf{u} \in \Omega_{\mathbf{U}}$ upper and lower bounds for $P(\mathbf{u})$ can be found by exploring only the extreme models in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$.*

Proof. The upper and lower expectations $\underline{E}[Z(\mathbf{U})] = \inf_{P \in \mathcal{S}_{\mathcal{M},\mathcal{D}}}\{E_P[Z(\mathbf{U})]\}$ and $\overline{E}[Z(\mathbf{U})] = \sup_{P \in \mathcal{S}_{\mathcal{M},\mathcal{D}}}\{E_P[Z(\mathbf{U})]\}$ of any real function $Z(\mathbf{U})$ in a set of probability mass functions $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ are achieved in extreme points of the convex envelope of $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ [15, Th. 7.9.3]. Note that, by definition of expectation, $E_P[Z(\mathbf{U})] = \sum_{\mathbf{u} \in \Omega_{\mathbf{U}}} Z(\mathbf{u})P(\mathbf{u})$.

Consider a function defined as

$$Z_{\mathbf{u}}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} = \mathbf{u}, \\ 0 & \text{otherwise,} \end{cases}$$

for all $\mathbf{x} \in \Omega_{\mathbf{U}}$. Then it holds that for all $\mathbf{u} \in \Omega_{\mathbf{U}}$,

$$P(\mathbf{u}) = E_P[Z_{\mathbf{u}}(\mathbf{U})].$$

On the other hand, it is clear that $\underline{E}[Z_{\mathbf{u}}(\mathbf{U})] \leq E_P[Z_{\mathbf{u}}(\mathbf{U})] \leq \overline{E}[Z_{\mathbf{u}}(\mathbf{U})]$ that, by denoting P_* and P^* the probability mass functions corresponding to $\underline{E}$ and $\overline{E}$ respectively, means that

$$E_{P_*}[Z_{\mathbf{u}}(\mathbf{U})] \leq E_P[Z_{\mathbf{u}}(\mathbf{U})] \leq E_{P^*}[Z_{\mathbf{u}}(\mathbf{U})] \Rightarrow P_*(\mathbf{u}) \leq P(\mathbf{u}) \leq P^*(\mathbf{u}),$$

where P_* and P^* are extreme points in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ according to Theorem 7.9.3 in [15]. □

The extreme models can be obtained from a proper reduction, as stated in the following:[2]

Theorem 3. *Let $\mathcal{M}$ be a partially-specified SCM, and $\mathcal{D}$ an M-compatible dataset. If $\mathcal{S}_{R(\mathcal{M},\mathcal{A}_{\mathbf{U}}),\mathcal{D}}$ contains a single model M, then M is an extreme SCM in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$.*

[2]This result follows directly from polytope theory, but we re-establish the result here to ensure consistent notation.

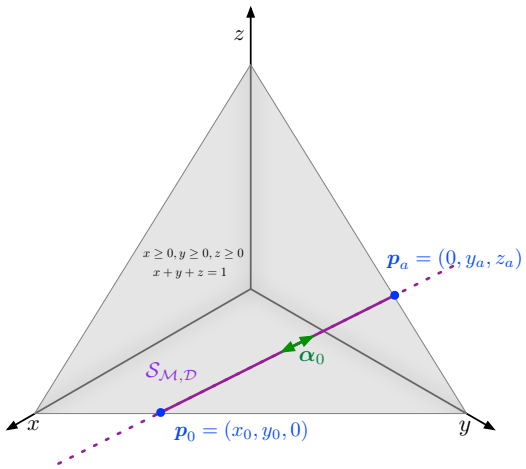


Figure 2: Visualization of the solution space. We restrict ourselves to solutions $(x, y, z) \in \mathbb{R}^3$ for easier visualization. $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ (purple line) is a linear subspace on the probability simplex (shaded area). The two extreme models in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ are named $\mathbf{p}_0$ and $\mathbf{p}_a$, respectively. $\boldsymbol{\alpha}_0$ defines an internal point on $\mathcal{S}_{\mathcal{M},\mathcal{D}}$, with an ϵ-ball of alternative models surrounding it. See Theorem 4 for context.

Proof. It follows from Theorem 1 that $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$. Now, assume M is not extreme in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. Then there exist two models M_1 and M_2 so that the distribution over the exogenous variables in M are given by a convex combination of the same distributions in the models M_1 and M_2. As probabilities are non-negative, it follows that whenever $P(\mathbf{u}) = 0$ for a specific $\mathbf{u} \in \Omega_{\mathbf{U}}$ in M, then the same must be true for M_1 and M_2 as well. Therefore, M_1 and M_2 have all the zero-allocations in their representations of $P(\mathbf{U})$ as M has, and $M_j \in \mathcal{S}_{R(\mathcal{M},\mathcal{A}_{\mathbf{U}}),\mathcal{D}}$ for $j = 1, 2$. This is a contradiction, since $\mathcal{S}_{R(\mathcal{M},\mathcal{A}_{\mathbf{U}}),\mathcal{D}}$ only contains a single model. It follows that M is extreme in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. □

The reverse is also true, and we will later use that to show completeness of our approach:

Theorem 4. *Let M be an extreme SCM in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. Then there exists a reduction $R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ s.t. M is the single model in $\mathcal{S}_{R(\mathcal{M},\mathcal{A}_{\mathbf{U}}),\mathcal{D}}$.*

Proof. Consider an SCM $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$, and focus for simplicity on one exogenous variable $U \in \mathbf{U}$ with states $\left\{u^{(j)}\right\}$. As $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$, it follows that $P_M(U)$ is a solution to Equation (4), a linear equation system with r rows and c columns; the system has rank $\rho \leq r < c$ (cf. Theorem 5). We will first show that an extreme model $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$ can have at most ρ non-zero elements. Assume instead that $P_M(U)$ has $\rho + k$ non-zero entries, $k > 0$; we drop the zero-elements from the equation-set. Note that since $P_M(U)$ is a solution to Equation (4), it can be written in the general form

$$P_M(U) = \mathbf{p}_0 + \boldsymbol{\alpha}^\top \mathbf{E}, \tag{7}$$

where $\mathbf{p}_0$ denotes the *particular solution*, and $\mathbf{E} = [\mathbf{e}_1 \dots \mathbf{e}_k]$ holds the basis vectors of the associated *null-space*, see also Figure 2. Since the null-space is a k-dimensional subspace of $\mathbb{R}^{\rho+k}$, there is a unique solution $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0$ of Equation (7) that defines $P_M(U)$. However, as $P_M(U)$ is non-zero in all $\rho+k$ dimensions, any model defined by an $\boldsymbol{\alpha}$ in the ϵ-ball around $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0$ is also a solution of Equation (4), and thus in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. In particular, define $\mathbf{1}$ to be the $(\rho + k)$-vector with all elements equal to 1, and consider both M_1 at $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 - \delta \cdot \mathbf{1}$ and M_2 at $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \delta \cdot \mathbf{1}$. Both models are in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ for sufficiently small $\delta > 0$, and since $M = \frac{1}{2}(M_1 + M_2)$, it follows that M is not extreme in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. This is a contradiction, thus $P_M(U)$ has at most ρ non-zero elements. Finally, by defining $\mathcal{A}_U$ to hold the states where $P_M(u^{(j)}) = 0$ (there are at least $c - \rho$ such states, we select $c - \rho$ of them arbitrarily if there are more), $R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ defines a full-rank linear equation system, and $\mathcal{S}_{R(\mathcal{M},\mathcal{A}_{\mathbf{U}}),\mathcal{D}} = \{M\}$. □

Example 4. *Consider again our running example. The solution space can be written as*

$$\mathcal{S}_{\mathcal{M},\mathcal{D}} = \{P(U) = \mathbf{p} \mid \mathbf{p} = \mathbf{p}_0 + \alpha \cdot \mathbf{e}\}$$

with $\mathbf{p}_0 = [0.323, 0.139, 0, 0.538]^\top$, $\mathbf{e} = [-1, +1, +1, -1]^\top$, *and* $\alpha \in [0, .323]$. *The two extreme points* $[0.323, 0.139, 0, 0.538]^\top$ *and* $[0, 0.462, 0.323, 0.215]^\top$ *are found by setting* $\alpha = 0$ *and* $\alpha = .323$, *respectively.*

Notice how we are able to recover all the extreme points of our running example also in Example 4. This is a consequence of Theorem 4, which states that all extreme points can be found by looking for the associated reduction sets $\mathcal{A}_{\mathbf{U}}$. Our strategy in the following will be to look for these reductions, and it follows from the theorem that we can do so without endangering the exactness of our approach.

4.2. SCMs with binary endogenous variables

The most general specification of a Markovian SCM $\mathcal{M}$ lets each endogenous variable Y have a single exogenous parent U with states that index all the possible deterministic structural equations from the set of Y's endogenous parents in $\mathcal{M}$ to Y. With this general specification, for a dataset $\mathcal{D}$, any model M that is compatible with $\mathcal{D}$ is contained in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$. For this most general SCM $\mathcal{M}$, the size of the state space of an exogenous variable U is determined by the size of the state spaces of its endogenous child variable Y and of Y's endogenous parents $\mathbf{X} = \mathrm{Pa}_Y \setminus \{U\}$, with $|\Omega_U| = |\Omega_Y|^{|\Omega_\mathbf{X}|}$. Letting $m = |\Omega_Y|^{|\Omega_\mathbf{X}|}$, define $\Omega_U = \{u_i\}_{i=0}^{m-1}$. A mapping between the possible functions f_{u_i} from $\mathbf{X}$ to Y and states in Ω_U then completes the definition of the structural function $f_Y(\mathbf{X}, U) = f_U(\mathbf{X})$ in $\mathcal{M}$.

Each structural function $f_Y(\mathbf{X}, U)$ of a model $\mathcal{M}$ is defined for the isolated model component $\mathbf{X} \rightarrow Y \leftarrow U$ independently of the remaining variables of the model. Determining the set of $\mathcal{K}(U)$ for exogenous U compatible with $\mathcal{D}$ in turn is independent of the credal sets of the remaining exogenous variables of the model. In the following, the scope of the discussion is a single model component $\mathbf{X} \rightarrow Y \leftarrow U$ with credal set $\mathcal{K}(U)$. Extending the results to general models $\mathcal{M}$ with $|\mathbf{U}| > 1$ is straight forward, repeating the process described for each exogenous variable U that is part of the model.

Assume for now that all endogenous variables have binary state spaces, such that $|\Omega_U| = 2^{2^n}$ with n the number of endogenous parents of endogenous child Y of exogenous U. A structured approach to defining a consistent mapping from u_i to f_{u_i} is defined as follows: For a specific ordering $(\mathbf{x}_i)_{i=1}^{2^n}$ of the 2^n states of the endogenous parents $\mathbf{X}$, the function indexed by u_i is given by the binary encoding of i to 2^n digits, such that the digit at position j corresponds to the output y of $f_{u_i}(\mathbf{x}_j)$. If $n = 2$, with the order of states for $\mathbf{X} = (X_1, X_2)$ as $((0,0),(0,1),(1,0),(1,1))$, u_0 is defined by binary string $0_{10} = 0000_2$ such that $f_Y(X_1, X_2, u_0) = 0$ for all four input states. For u_3, $3_{10} = 0011_2$ defines $f_Y(X_1, X_2, u_3) = X_1$, i.e. for the first two states $(0,0),(0,1)$ of X_1, X_2, the output is 0 and for the next two states $(1,0),(1,1)$ the output is 1. Tables 1 and 2 detail the complete mapping for the case where Y has one endogenous parent X, and where Y has two endogenous parents X_1, X_2, respectively.

Given this definition of $f_Y(\mathbf{X}, U)$, the credal set $\mathcal{K}(U)$ is now defined by

X	Y	U				$\tilde{P}(Y\|X)$
		u_0	u_1	u_2	u_3	
0	0	1	1	0	0	$\tilde{P}(Y=0\|X=0)=P(u_0)+P(u_1)$
0	1	0	0	1	1	$\tilde{P}(Y=1\|X=0)=P(u_2)+P(u_3)$
1	0	1	0	1	0	$\tilde{P}(Y=0\|X=1)=P(u_0)+P(u_2)$
1	1	0	1	0	1	$\tilde{P}(Y=1\|X=1)=P(u_1)+P(u_3)$
$f_U(X)=$		0	X	$\overline{X}$	1	

Table 1: A canonical specification of Ω_U when endogenous child Y of U has one endogenous parent X. For columns with header u_i, a 1 indicates that the function $f_{u_i}(X) = Y$ as defined in the respective column is consistent with the pair of values for X, Y as listed in the respective row, i.e. that Y takes on its value with probability 1 given the value of X and U. Conversely, a 0 indicates 0 probability of the value combination in question. While the complete table shows probabilities of all state combinations, this can be translated into the corresponding binary representation column-wise by selecting the Y-value of probability 1 for each pair of consecutive rows. The bottom row summarises the functions defined. The rightmost column shows the resulting linear equations defining the credal set $\mathcal{K}(U)$.

the following linear system:

$$\sum_{u \in \Omega_U^{j,k}} P(u) = \tilde{P}(Y = y_k | \mathbf{X} = \mathbf{x}_j), \quad y_k \in \Omega_Y, \mathbf{x}_j \in \Omega_{\mathbf{X}}, \tag{8}$$

with $u_i \in \Omega_U^{j,k}$ if position j in the binary encoding of i equals y_k (See the rightmost column in Table 1 for an example). Note that, since we are considering binary endogenous variables, $\Omega_Y = \{0, 1\}$ and $\Omega_{\mathbf{X}} = \{0, 1\}^n$. This linear system is underdetermined with 2^{2^n} unknowns, such that there are infinitely many solution models M for $\mathcal{M}$, for which $\mathcal{D}$ is M-compatible. Now, consider reductions R for which the resulting model $\mathcal{M}' = R(\mathcal{M}, \mathcal{A}_U)$ has a single solution M. The following theorem bounds the size of the domain of an exogenous variable U' of a reduced model $\mathcal{M}'$ for which the linear system has a unique solution:

X_1	X_2	Y	U																$P(Y\|X_1, X_2)$
			u_0	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}	
0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	$P(Y=0\|X_1=0, X_2=0)$
0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	$P(Y=1\|X_1=0, X_2=0)$
0	1	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	$P(Y=0\|X_1=0, X_2=1)$
0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	$P(Y=1\|X_1=0, X_2=1)$
1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	$P(Y=0\|X_1=1, X_2=0)$
1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	$P(Y=1\|X_1=1, X_2=0)$
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	$P(Y=0\|X_1=1, X_2=1)$
1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	$P(Y=1\|X_1=1, X_2=1)$
$f_U(X_1, X_2) =$			0	X_1 $\wedge$ X_2	X_1 $\wedge$ $\overline{X}_2$	X_1	$\overline{X}_1$ $\wedge$ X_2	X_2	X_1 $\oplus$ X_2	X_1 $\vee$ X_2	$\overline{X}_1$ $\wedge$ $\overline{X}_2$	X_1 $\oplus$ $\overline{X}_2$	$\overline{X}_2$	X_1 $\vee$ $\overline{X}_2$	$\overline{X}_1$	$\overline{X}_1$ $\vee$ X_2	$\overline{X}_1$ $\vee$ $\overline{X}_2$	1	

Table 2: A canonical specification of Ω_U when endogenous child Y of U has two endogenous parents X_1 and X_2. The table is read analogously to Table 1. The probabilities in the rightmost column are again the sum of the probabilities of all u_i for which the respective column evaluates to 1.

Theorem 5. *Let $\mathcal{M}$ be a partially-specified Markovian SCM with binary endogenous variables, and $\mathcal{D}$ an M-compatible dataset. For an exogenous variable U with endogenous child variable Y in $\mathcal{M}$, let n denote the number of endogenous parents of Y. If a reduction R is applied to $\mathcal{M}$ returning $\mathcal{M}'$ such that the resulting linear system defining $\mathcal{K}(U')$ has exactly one solution M for $\mathcal{D}$, then $|\Omega_{U'}| \leq 2^n + 1$, where U' is the image of U under R.*

Proof. The equation $\sum_{i=0}^{2^{2^n}-1} P(u_i) = 1$ along with $\{P(Y = 0|\mathbf{X} = \mathbf{x}_j)\}_{j=1}^{2^n}$ ($\{P(Y = 1|\mathbf{X} = \mathbf{x}_j)\}_{j=1}^{2^n}$ being dependent given $\sum_i P(u_i) = 1$) together form a system of $2^n + 1$ independent linear equations. For a number of unknowns after reduction $|\Omega_{U'}| > 2^n + 1$, the system solution space is infinite. □

Such a model reduction R, for which $|\mathcal{K}(U')| = 1$, effectively reduces the number of unknowns in $\{P(u_i)\}_{i\in\Omega_U}$ by fixing $2^{2^n} - (2^n + 1)$ of these

probabilities to be 0. If the unique solution of the resulting linear system given data $\mathcal{D}$ respects $P(u_i) \geq 0$ and $\sum_i P(u_i) = 1$, then this solution is a model $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$, for which $q_M \in q_{\mathcal{M},\mathcal{D}}$. From Theorems 3 and 4 it follows that M is an extreme model, and that all extreme models in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$ belong to a subspace $\mathcal{M}' = R(\mathcal{M}, \mathcal{A}_{\mathbf{U}})$ with $|\mathcal{A}_{\mathbf{U}}| = 2^{2^n} - (2^n + 1)$.

4.3. Satisfiability

Algorithm 1 outlines the steps of the process of finding fully-specified SCMs given a partially-specified model $\mathcal{M}$ and a dataset $\mathcal{D}$, where Line 4 corresponds to finding a reduction such that the resulting model has a unique solution. Now, Theorem 5 states that such a reduction R may reduce the complexity of the model to be solved significantly in replacing variable U, for which $|\Omega_U| = 2^{2^n}$, by U' with $|\Omega_{U'}| \leq 2^n + 1$. However, the space of possible reductions consists of a total of $\binom{2^{2^n}}{2^n+1}$ distinct R's, for which U' has the required domain size $2^n + 1$. Moreover, out of this set of reduced models, only a potentially small subset will be consistent with a given dataset $\mathcal{D}$. Thus, this search is not straightforward when n grows.

A connection to the satisfiability problem is introduced next. Defining a new set of variables $\{z_i\}_{i=0}^{m-1}$ such that

$$z_i = \begin{cases} \text{True} & \text{if } P(u_i) > 0, \\ \text{False} & \text{if } P(u_i) = 0, \end{cases}$$

a Conjunctive Normal Form (CNF) formula may be formed from the left hand side expressions of the equation set given by Equation (8). For each sum $\sum_{u \in \Omega_U^{j,k}} P(u)$, a disjunction clause over the variables in the corresponding set $\{z_i : i \text{ s.t. } u_i \in \Omega_U^{j,k}\}$ is added to the CNF formula, such that the complete

Algorithm 1 Solution enumeration

input: $\mathcal{M}$ (partially-specified SCMs), $\mathcal{D}$ (endogenous dataset), N (number of runs)

output: $\mathcal{S} = \{M_1, M_2, \ldots, M_N\}$ (set of fully-specified SCMs)

1: $\mathcal{S} \leftarrow \emptyset$
2: **for** $i \in \{1, \ldots, N\}$ **do**
3: **for** $U \in \mathbf{U}$ **do**
4: Find a $\mathcal{A}_U \subset \Omega_U$ s.t. $\mathcal{M}' = R(\mathcal{M}, \mathcal{A}_U)$ has a single solution for U.
5: $P'(U) \leftarrow$ solve U in $\mathcal{M}'$ given the data $\mathcal{D}$.
6: **end for**
7: $M_i \leftarrow$ build an SCM with $\{P'(U)\}_{U \in \mathbf{U}}$.
8: $\mathcal{S} \leftarrow \mathcal{S} \cup \{M_i\}$
9: **end for**
10: **return** $\mathcal{S}$

Algorithm 2 Inference

input: $\mathcal{S}$ (set of fully-specified SCMs), q (causal or counterfactual query)

output: bounds of q

1: $\mathbf{q} \leftarrow \emptyset$
2: **for** $M \in \mathcal{S}$ **do**
3: $\mathbf{q} \leftarrow \mathbf{q} \cup \{q_M\}$
4: **end for**
5: **return** $(\min(\mathbf{q}), \max(\mathbf{q}))$

formula is the conjunction of all $2 \cdot 2^n$ disjunction clauses. For $n = 1$, the complete equation set as shown in the rightmost column of Table 1 corresponds to the CNF formula $g(\mathbf{z}) = (z_0 \vee z_1) \wedge (z_2 \vee z_3) \wedge (z_0 \vee z_2) \wedge (z_1 \vee z_3)$. Now, in order for any subset $\{P(u_i)\}_{i \in \mathcal{I}}, |\mathcal{I}| = 2^n + 1$, to solve the linear system given by Equation (8), the variables in $\{z_i\}_{i \in \mathcal{I}}$ set to True must be a solution to the corresponding CNF formula $g(\mathbf{z})$. If not, $P(u) = 0 \; \forall u \in \Omega_U^{j,k}$ for some pair of values $\mathbf{x}_j, y_k$, which in general will violate Equation (8). Thus it is only necessary to consider $\mathcal{I}$ such that $g(\mathbf{z})$ is satisfied, to be possible model solutions of the linear system.

Thus, Line 4 of Algorithm 1 may be accomplished by first finding a set

$\{z_i\}_{i\in\mathcal{I}}$ that solves $g(\mathbf{z})$, and only then solving the linear system, by setting all $\{P(u_j)\}_{j=0}^{m-1} \setminus \{P(u_i)\}_{i\in\mathcal{I}}$ to 0 and then inverting the resulting $(2^n+1)\times(2^n+1)$ square matrix. If this unique solution satisfies $\sum_{i\in\mathcal{I}} P(u_i) = 1$ and $P(u_i) \geq 0, \forall i \in \mathcal{I}$, the solution corresponds to a fully-specified model $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$, of which q_M lies in $q_{\mathcal{M},\mathcal{D}}$ to be approximated. Algorithm 2 details the approximation of $q_{\mathcal{M},\mathcal{D}}$ given a set of fully-specified models found by Algorithm 1.

For $n = 1$, all subsets of $\{P(u_i)\}_{i=0}^{3}$ of size $2^1 + 1 = 3$ correspond to subsets of $\{z_i\}_{i=0}^{3}$ that satisfy $g(\mathbf{z})$, and the 3 equations given by Equation (8) may be solved for each of the $\binom{2^{2^1}}{2^1+1} = \binom{4}{3} = 4$ possible subsets, such that a solution model $M \in \mathcal{S}_{\mathcal{M},\mathcal{D}}$ is found if the solution corresponds to a probability distribution.

Example 5. *There are four possible reductions $R(\mathcal{M}, u_i)$ for $i \in \{0,1,2,3\}$, for variable U of model $\mathcal{M}$ of Figure 1. With data $\mathcal{D}$ of Figure 1, the solution set $\mathcal{K}(U)$ for $R(\mathcal{M}, u_2)$ is as shown in Example 3. The reduction $R(\mathcal{M}, u_0)$ returns a model with the solution set $\mathcal{K}(U) = \{[P(u_1) = 0.462, P(u_2) = 0.323, P(u_3) = 0.215]\}$, while both $R(\mathcal{M}, u_1)$ and $R(\mathcal{M}, u_3)$ are models with empty solution sets for $\mathcal{D}$.*

For $n = 2$, it is no longer the case that all of the size $2^2 + 1 = 5$ subsets of $\{P(u_i)\}_{i=0}^{15}$ satisfies $g(\mathbf{z})$. The total solution space of $\binom{16}{5} = 4368$ possible solutions may still be searched exhaustively, tested for satisfiability against $g(\mathbf{z})$, then tested by solving the equation system for possible solution models. Note that by Theorems 3, 4 and 5 all extreme distributions $P(U)$ will be explored if these solutions are searched exhaustively, and the interval returned will be the true query interval. For $n = 3$ however, $\binom{2^{2^n}}{2^n+1} = \binom{256}{9} \approx 10^{16}$,

and a complete search is no longer feasible. Thus, a heuristics based search approach is described next, to allow faster retrieval of models in $\mathcal{S}_{\mathcal{M},\mathcal{D}}$, based on ensuring satisfiability of the CNF formula.

4.4. Heuristic-based solution search

Now, instead of testing every possible subset among the $\binom{2^{2^n}}{2^n+1}$ possibilities, the approach presented here will generate sets $\{z_i\}_{i\in\mathcal{I}}$ of size $|\mathcal{I}| = 2^n+1$ in such a way that 1) the CNF formula is guaranteed to be satisfied and 2) the solution for $\{P(u_i)\}_{i\in\mathcal{I}}$ is more likely to be a probability distribution than a randomly sampled subset. Each $P(u_i)$ contributes to exactly one of $\tilde{P}(Y = 0|\mathbf{X} = \mathbf{x}_j)$ or $\tilde{P}(Y = 1|\mathbf{X} = \mathbf{x}_j)$. Thus for a pair of distribution clauses, any z_i not part of one clause is part of the other. This can be seen in Figure 3, which shows the CNF disjunction clauses when Y has two endogenous parents X_1, X_2.

Now, this relationship between the clauses within a distribution may be exploited in order to generate subsets of variables that satisfy the CNF formula. Specifically, for any collection of exactly one clause per configuration of endogenous parents $\mathbf{x}$, selecting a set of two or more variables such that no two variables of the set appear in the same clause will satisfy the formula, due to the inverse symmetry across distributions. See Figure 3 for an example, where both sets $\{z_7, z_{11}, z_{13}, z_{14}\}$ and $\{z_3, z_{13}, z_{14}\}$ are identified as CNF-solutions for $n = 2$ over the clauses for $\tilde{P}(Y = 0|X_1 = x_1, X_2 = x_2), \forall x_1 \in \Omega_{X_1}, x_2 \in \Omega_{X_2}$.

Furthermore, this can be done systematically such that distinct partial solutions are generated across all 2^n possible sets of clauses. The approach generates partial CNF-solutions of size m, where $2 \leq m \leq 2^n$, but any

CNF disjunction clauses	$P(Y\|X_1, X_2)$
$z_0 \vee z_1 \vee z_2 \vee z_3 \vee z_4 \vee z_5 \vee z_6 \vee z_7$	$P(Y=0\|X_1=0, X_2=0)$
$z_8 \vee z_9 \vee z_{10} \vee z_{11} \vee z_{12} \vee z_{13} \vee z_{14} \vee z_{15}$	$P(Y=1\|X_1=0, X_2=0)$
$z_0 \vee z_1 \vee z_2 \vee z_3 \vee z_8 \vee z_9 \vee z_{10} \vee z_{11}$	$P(Y=0\|X_1=0, X_2=1)$
$z_4 \vee z_5 \vee z_6 \vee z_7 \vee z_{12} \vee z_{13} \vee z_{14} \vee z_{15}$	$P(Y=1\|X_1=0, X_2=1)$
$z_0 \vee z_1 \vee z_4 \vee z_5 \vee z_8 \vee z_9 \vee z_{12} \vee z_{13}$	$P(Y=0\|X_1=1, X_2=0)$
$z_2 \vee z_3 \vee z_6 \vee z_7 \vee z_{10} \vee z_{11} \vee z_{14} \vee z_{15}$	$P(Y=1\|X_1=1, X_2=0)$
$z_0 \vee z_2 \vee z_4 \vee z_6 \vee z_8 \vee z_{10} \vee z_{12} \vee z_{14}$	$P(Y=0\|X_1=1, X_2=1)$
$z_1 \vee z_3 \vee z_5 \vee z_7 \vee z_9 \vee z_{11} \vee z_{13} \vee z_{15}$	$P(Y=1\|X_1=1, X_2=1)$

Figure 3: The CNF formula disjunction clauses over variables $\{z_i\}_{i=0}^{15}$ when Y has two endogenous parents X_1, X_2. Each pair of consecutive rows make up a single conditional distribution. Conditional probabilities are shown in the rightmost column, with left hand clauses corresponding to conditional probabilities for $Y = 0$ given different values for parents X_1, X_2, and right hand clauses correspond to conditional probabilities for $Y = 1$. Over the set of left hand clauses, two examples of variable sets are circled that both satisfy the full formula. Both sets $\{z_7, z_{11}, z_{13}, z_{14}\}$ (in red) and $\{z_3, z_{13}, z_{14}\}$ (in blue) are such that exactly one of the variables is present in each clause, while the remaining variables must be present in the corresponding right hand clause.

choice of additional variables may be included in the partial set not affecting satisfiability, in order to find complete $2^n + 1$-size CNF-solutions. While the approach so far guarantees that all solutions $\mathcal{I}$ searched are such that $\{z_i\}_{i\in\mathcal{I}}$ solves the CNF-formula, the corresponding unique solution to (8) will most of the times not correspond to a probability distribution. Thus, in order to focus the search towards the probability simplex, the approach may consider only some of the 2^n subsets of clauses to build the partial solutions across. Specifically, clauses may be selected according to their corresponding probability: For each distribution, choose the clause of lowest probability, and for this set of lowest probability clauses, build partial solutions. An

example is shown in Figure 4.

CNF disjunction clauses	$P(Y\|X_1, X_2)$
$z_0 \vee z_1 \vee z_2 \vee z_3 \vee z_4 \vee z_5 \vee z_6 \vee z_7$ $z_8 \vee z_9 \vee z_{10} \vee z_{11} \vee z_{12} \vee z_{13} \vee z_{14} \vee z_{15}$	$P(Y=0\|X_1=0, X_2=0) = 0.95$ $P(Y=1\|X_1=0, X_2=0) = 0.05$
$z_0 \vee z_1 \vee z_2 \vee z_3 \vee z_8 \vee z_9 \vee z_{10} \vee z_{11}$ $z_4 \vee z_5 \vee z_6 \vee z_7 \vee z_{12} \vee z_{13} \vee z_{14} \vee z_{15}$	$P(Y=0\|X_1=0, X_2=1) = 0.78$ $P(Y=1\|X_1=0, X_2=1) = 0.22$
$z_0 \vee z_1 \vee z_4 \vee z_5 \vee z_8 \vee z_9 \vee z_{12} \vee z_{13}$ $z_2 \vee z_3 \vee z_6 \vee z_7 \vee z_{10} \vee z_{11} \vee z_{14} \vee z_{15}$	$P(Y=0\|X_1=1, X_2=0) = 0.07$ $P(Y=1\|X_1=1, X_2=0) = 0.93$
$z_0 \vee z_2 \vee z_4 \vee z_6 \vee z_8 \vee z_{10} \vee z_{12} \vee z_{14}$ $z_1 \vee z_3 \vee z_5 \vee z_7 \vee z_9 \vee z_{11} \vee z_{13} \vee z_{15}$	$P(Y=0\|X_1=1, X_2=1) = 0.39$ $P(Y=1\|X_1=1, X_2=1) = 0.61$

Figure 4: The CNF formula disjunction clauses sorted by probability distributions. Clauses to the left correspond to lowest probabilities. Choosing partial subsets over these clauses bias the selected variable set such that variables appear more often in high-probability clauses. Circled in red is the size 4 partial CNF-solution $\mathcal{Z}_1 = \{z_1, z_2, z_7, z_{11}\}$ and circled in blue is the size 3 partial CNF-solution $\mathcal{Z}_2 = \{z_1, z_2, z_{15}\}$. Indeed, for each clause on the right hand side corresponding to higher probabilities, $|\mathcal{Z}_1| - 1 = 3$ of the variables in set $\mathcal{Z}_1$ appear, as does $|\mathcal{Z}_2| - 1 = 2$ of the variables in set $\mathcal{Z}_2$.

This approach biases the search towards solutions that will have more non-zero components in the equations that sum to probabilities > 0.5, and similarly fewer non-zero components in equations that sum to < 0.5. The partial solutions still need to be expanded to $2^n + 1$ size, which could similarly be approached by favouring remaining variables that appear most often in higher probability clauses. Note that while a complete search of all extreme solutions satisfying the CNF solution is guaranteed to retrieve the full interval, restricting the search as described according to observed probabilities no longer guarantees completeness. However, experiments show that searching by this heuristic ensures good interval approximations when model complexity renders exhaustive search no longer feasible, see Section 5.

Finally, note the special case in which $\tilde{P}(Y = y_k|\mathbf{X} = \mathbf{x}_j) = 0$ for one or more pairs of values $(y_k, \mathbf{x}_j) \in \Omega_Y \times \Omega_{\mathbf{X}}$. If a conditional probability equals 0, it forces every component of the corresponding sum $\sum_{u \in \Omega_U^{k,j}} P(u)$ to be 0, and by such reduces the number of equations in the system by 1. Thus solution sets $\mathcal{I}$ are now to be of size 2^n. This also affects the CNF formula in the following way: The disjunction clause corresponding to the equation in question becomes negated, ensuring no variable part of that clause can evaluate to True while solving the formula. This similarly extends if more than one conditional equals 0, reducing both the number of equations and the size of the set of variables to choose from in search of a solution.

In order for the heuristics-based search to deal with such special cases most efficiently, for any equation $\sum_{u \in \Omega_U^{k,j}} P(u) = 0$ part of the system, the solutions searched are restricted to those containing exactly one variable z_i for which $u_i \in \Omega_U^{k,j}$, while keeping the equation as part of the system, such that $P(u_i)$ is solved to be 0 as part of a now still unique solution to the $2^n + 1$ equations of the linear system.

4.5. SCMs with non-binary endogenous variables

The approach presented is so far detailed for SCMs with binary endogenous variables only. In this section, the scope is generalized to allow for discrete endogenous variables of cardinality larger than two.

Again, a solution search is defined independently for each U in the partial model $\mathcal{M}$. Consider an exogenous variable U with endogenous child Y and let $\mathbf{X}$ be the set of endogenous parents of Y. Assume $|\Omega_{\mathbf{X}}| = p$, $|\Omega_Y| = q$, such that $|\Omega_U| = q^p$. A standard mapping from values u_i in Ω_U to functions f_{u_i} is defined as follows:

The function indexed by u_i is now given by i expressed in the base-q numeral system, to p digits. Assuming a given ordering of values $(\mathbf{x}_j)_{j=1}^{p}$ for variable $\mathbf{X}$, the digit at position j in the base-q encoding of i corresponds to the output y of $f_{u_i}(\mathbf{x}_j)$. If $p = 2, q = 4$ and $\{0, 1\}$ is the ordered set of states for $\mathbf{X}$, u_0 is defined by $0_{10} = 00_4$ such that $f_Y(\mathbf{X}, u_0) = 0$ for both states of input $\mathbf{X}$. For u_7 as another example, $7_{10} = 13_4$ defines

$$f_Y(\mathbf{X}, u_7) = \begin{cases} 1 & \text{if } \mathbf{X} = 0, \\ 3 & \text{if } \mathbf{X} = 1. \end{cases}$$

Table 3 details the complete mapping from u_i to f_{u_i} when $q = 4$ and $p = 2$, with $|\Omega_U| = 16$.

X	Y	U																$P(Y\|X)$
		u_0	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}	
0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	$P(Y=0\|X=0)$
0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	$P(Y=1\|X=0)$
0	2	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	$P(Y=2\|X=0)$
0	3	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	$P(Y=3\|X=0)$
1	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	$P(Y=0\|X=1)$
1	1	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	$P(Y=1\|X=1)$
1	2	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	$P(Y=2\|X=1)$
1	3	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	$P(Y=3\|X=1)$
i (base 4)		00	01	02	03	10	11	12	13	20	21	22	23	30	31	32	33	

Table 3: A canonical specification of Ω_U when endogenous child Y of U, with values in $\{0, 1, 2, 3\}$, has endogenous parent X with values in $\{0, 1\}$. The table is read analogously to Tables 1 and 2. As before, probability equations are read from the table as $\tilde{P}(Y = 0|X = 0) = P(u_0) + P(u_1) + P(u_2) + P(u_3)$ for the first row, etc. The last row includes the base $q = 4$ representation to $p = 2$ digits of corresponding u_i as consistent with the table.

With this extended definition for $f_Y(X, U)$, the credal set $\mathcal{K}(U)$ analo-

gously follows as:

$$\sum_{u \in \Omega_U^{j,k}} P(u) = \tilde{P}(Y = y_k | X = x_j), \quad y_k \in \Omega_Y, x_j \in \Omega_X \tag{9}$$

with $\Omega_U^{j,k} = \{u_i \in \Omega_U, i = 0, \ldots, |U|-1$ s.t. position j in the base-q encoding of i equals $y_k\}$. This is now a linear system in q^p unknowns.

Theorem 5 is updated below, and bounds, for any partial SCM $\mathcal{M}$ with discrete endogenous variables of finite cardinality, the size of the domain of an exogenous variable U' of a reduced model $\mathcal{M}'$ after reduction $R(\mathcal{M}, \mathcal{A}_U)$ on $\mathcal{M}$, where the resulting linear system has a unique solution.

Theorem 6. *Let $\mathcal{M}$ be a partially-specified Markovian SCM with discrete endogenous variables, and $\mathcal{D}$ an M-compatible dataset. For an exogenous variable U with endogenous child variable Y with endogenous parents $\mathbf{X}$ in $\mathcal{M}$, let $p = |\Omega_{\mathbf{X}}|$ and $q = |\Omega_Y|$. If a reduction R is applied to $\mathcal{M}$ returning $\mathcal{M}'$ such that the resulting linear system defining $\mathcal{K}(U')$ has exactly one solution M for $\mathcal{D}$, then $|\Omega_{U'}| \leq p \cdot (q-1) + 1$, where U' is the image of U under R.*

Proof. The equation $\sum_{i=0}^{q^p-1} P(u_i) = 1$ along with $\{\{P(Y = y_k | \mathbf{X} = \mathbf{x}_j)\}_{k=1}^{q-1}\}_{j=1}^{p}$ ($\{P(Y = y_q | \mathbf{X} = \mathbf{x}_j)\}_{j=1}^{p}$ being dependent given $\sum_i P(u_i) = 1$) together form a system of $p \cdot (q-1) + 1$ independent linear equations. For a number of unknowns after reduction $|\Omega_{U'}| > p \cdot (q-1) + 1$, the system solution space is infinite. □

With the number of independent equations in the linear system now at $p \cdot (q-1) + 1$, the search for extreme models is now restricted to reductions R such that the image U' of U under R has $|\Omega_{U'}| \leq p \cdot (q-1) + 1$. The space of possible reductions is of size $\binom{q^p}{p \cdot (q-1)+1}$. Generalizing the heuristic search

for extreme models in order to allow for non-binary endogenous variables is straightforward, using the same techniques as before.

The mapping from $\mathcal{K}(U)$ to corresponding CNF formula $g(\mathbf{z})$ is equivalent to that described for SCMs with binary endogenous variables (Section 4.3). As an example, the $\mathcal{K}(U)$ defined in Table 3 can be translated into the following CNF formula: $g(\mathbf{z}) = (z_0 \vee z_1 \vee z_2 \vee z_3) \wedge (z_4 \vee z_5 \vee z_6 \vee z_7) \wedge (z_8 \vee z_9 \vee z_{10} \vee z_{11}) \wedge (z_{12} \vee z_{13} \vee z_{14} \vee z_{15}) \wedge (z_0 \vee z_4 \vee z_8 \vee z_{12}) \wedge (z_1 \vee z_5 \vee z_9 \vee z_{13}) \wedge (z_2 \vee z_6 \vee z_{10} \vee z_{14}) \wedge (z_3 \vee z_7 \vee z_{11} \vee z_{15})$, mapping each $P(u_i)$ to z_i for all $0 \leq i \leq 15$. Two example CNF solutions for this formula are shown in Figure 5.

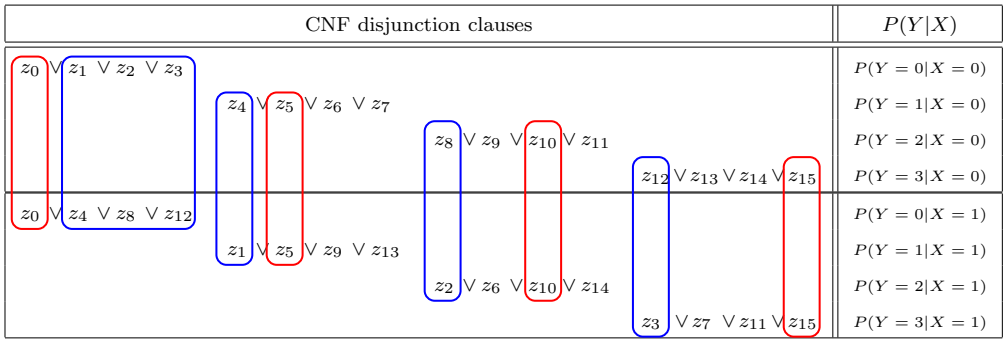


Figure 5: The CNF formula disjunction clauses corresponding to Table 3. Circled in red is the size 4 partial CNF solution $\mathcal{Z}_1 = \{z_0, z_5, z_{10}, z_{15}\}$, circled in blue is the size 6 partial CNF-solution $\mathcal{Z}_2 = \{z_1, z_2, z_3, z_4, z_8, z_{12}\}$. Removing any variable from either of these two sets will result in a new set that is no longer a CNF solution. To reach the required size $p \cdot (q-1) + 1 = 2 \cdot (4-1) + 1 = 7$, additional variables may be added before solving for possible distributions in 7 non-zero $P(u_i)$.

In order to increase the probability of a CNF solution corresponding to a probability distribution when solving the resulting equation system, as before the CNF solutions are generated after sorting the clauses according to the

corresponding observed probability. As each distribution now gives q clauses, one or more of these may be restricted to contain at most one z_i at value True, while remaining z_i will be spread across remaining clauses. By restricting the number of non-zero components $p(u_i)$ in the equations summing to the lowest probabilities, the resulting system solution is more likely to respect $p(u_i) \geq 0$. Generating CNF solutions under such constraints limits the search space as q and p grow, allowing for approximate query bounds where exhaustive search is no longer feasible.

Appendix A details the CNF solution generation technique used for model search in the experiments.

5. Experiments

5.1. Benchmark Generation

For validation, we consider a benchmark of 972 randomly generated SCMs with the three inverted tree topologies shown in Figure 6. The leaf node, denoted as Y, is an endogenous variable that can be either binary or ternary and has a set of binary endogenous independent parents, denoted as Pa_Y. Thus, each model configuration can be identified by the number of parents $|\mathrm{Pa}_Y| \in \{1, 2, 3\}$, and by the cardinality of the leaf $|\Omega_Y| \in \{2, 3\}$. All the models considered are Markovian, meaning that the association between exogenous and endogenous variables is one-to-one. Initially, all models are defined as canonical. However, to introduce variety in the complexity, some of the SEs are made non-canonical by randomly dropping certain states in the exogenous variables. After randomly initializing the exogenous distributions, an M-compatible dataset of 1000 instances is sampled. The reasons

for selecting these models are twofold: First, since learning is performed independently for each exogenous variable, the computational complexity does not increase for larger models with the same maximum in-degree. Second, for comparison purposes, it is crucial to compute the exact bounds for any query, which may not be feasible in models with a higher in-degree or greater variable cardinality.

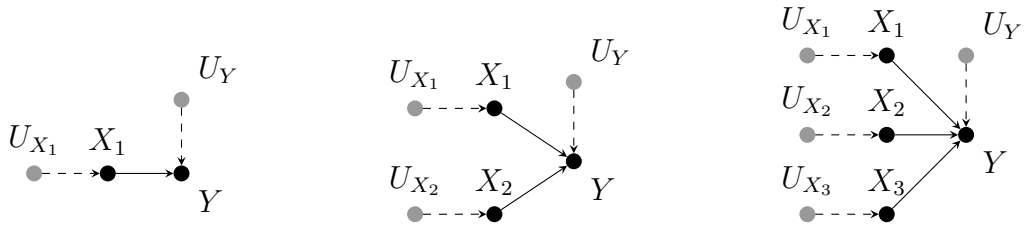


Figure 6: Graphs of the SCMs considered in the experimentation.

5.2. Experimental Setup

We compare our method (DCCC) against three alternatives: the exact linear programming approach proposed in [7]; EMCC [8] using a fixed number of EM iterations set to 50, 100, or 150; and the Gibbs sampling approach proposed in [5] with a burn-in phase of 100 iterations and computation of the full 100% credible interval. For each model and method, the learning process is evaluated using a varying number of generated solutions (parameter N from Algorithm 1), ranging from 1 to 70. Recall that a solution corresponds to a fully specified SCM. For the models with ($|\mathrm{Pa}_Y| = 1, |\Omega_Y| = 3$) or with ($|\mathrm{Pa}_Y| = 2, |\Omega_Y| = 2$), DCCC is applied exhaustively, whereas heuristic search is used for the more complex models. Different queries are considered, specifically PS and PN, with Y as the effect and one of the parents as the

cause. The computation time measured includes both the generation of N solutions and the inference required to compute one of the aforementioned queries. The approximation error is quantified using the RMSE (*root mean squared error*) with respect to the exact bounds.

To complement the results of the main experiments as described so far, some results from running DCCC heuristic search on more complex models are included at the end of the section. The setup for these additional experiments is simplified in order to ensure feasible performance evaluation, the query considered now being $P(Y_{X_1=1} = 1|Y = 0, X_1 = 0)$ and the model structure that of the leftmost graph in Figure 6 only. These experiments are performed using random generation of empirical distributions $\tilde{P}$ directly. While more restricted, the results are included to indicate potential performance of DCCC heuristic search as model complexity grows out of scope of comparable methods. As the interval approximations returned by DCCC are always contained within the true interval, these results are presented in terms of percentage of interval retrieved.

The benchmark and Python code for replicating the experiments are available in a dedicated GitHub repository[3].

5.3. Results Analysis

Figure 7 shows the average computation time in seconds (i.e., learning and inference times) for an increasing number of generated solutions, N. The computation was done in a computer cluster made of 1024 cores (AMD EPYC 7542 32-Core Processor) where each experiment was executed sequentially in

[3] `https://github.com/PGM-Lab/2025-IJAR-DCCC`

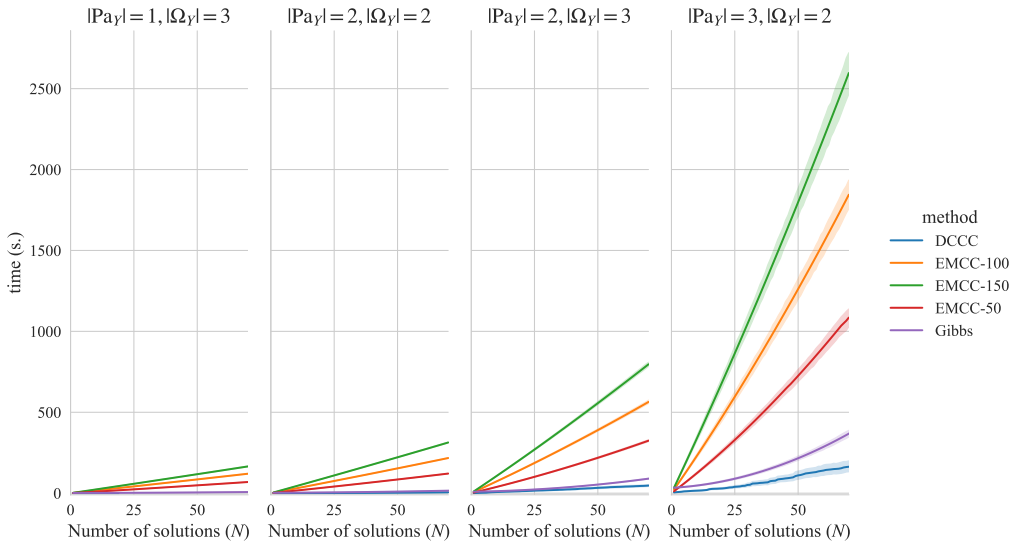


Figure 7: Average computation time for generating N solutions.

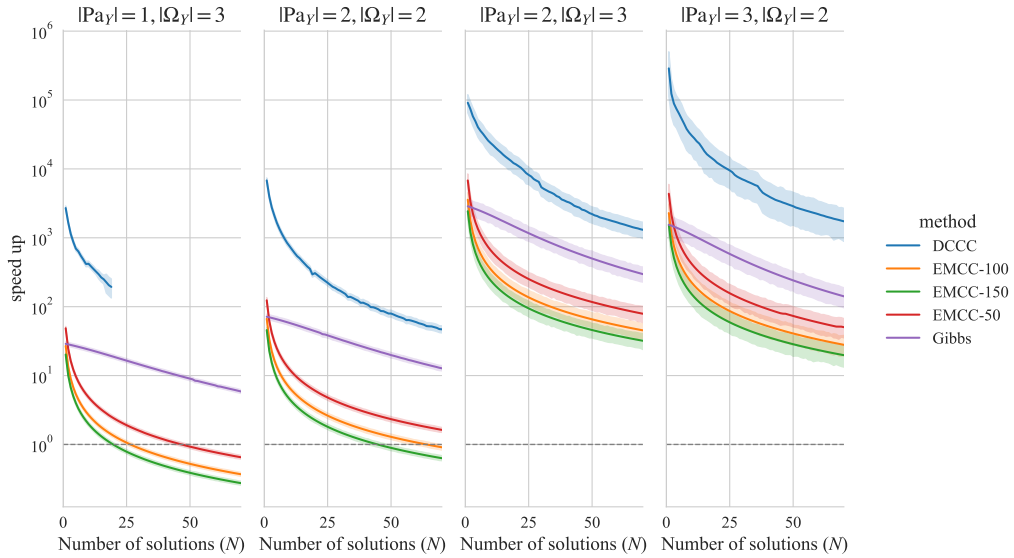


Figure 8: Average speed up for generating N solutions w.r.t the exact method. The dashed line indicates a speed up of 1.

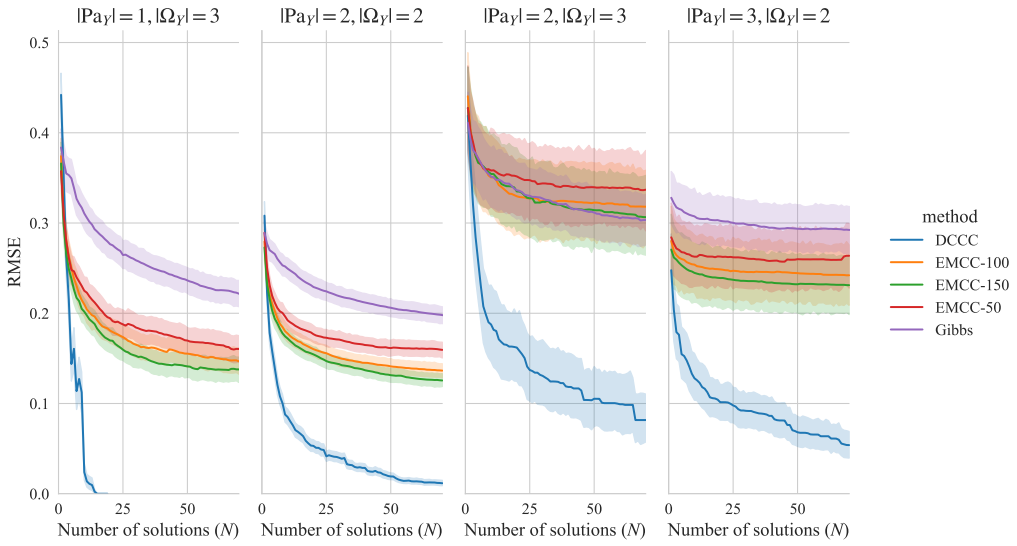


Figure 9: Error with respect the exact bounds vs. number of solutions N.

a single core. In all cases, DCCC proves to be the most efficient approach for generating each solution, a trend that becomes even more pronounced in the three-parent case. Additionally, in the one-parent case, the series for DCCC does not span the entire x-axis. This is because all possible reductions are analyzed, leading the algorithm to stop early.

$\lvert \mathrm{Pa}_Y \rvert$	$\lvert \Omega_Y \rvert$	Time (sec.)
1	3	43 s.
2	2	199 s.
2	3	$2.7 \cdot 10^4$ s.
3	2	$1.5 \cdot 10^5$ s.

Table 4: Computation time for exact method, compare to Figure 7.

Table 4 reports the computation time of the exact method, which varies

significantly across topologies. The last two values are extremely large and are omitted from Figure 7 to improve readability. Instead, Figure 8 presents the speed up achieved by each method relative to the exact method, demonstrating that our method is the fastest.

Beyond efficiency, DCCC also achieves the lowest error levels for a given number of visited solutions. This is illustrated in Figure 9, which presents the RMSE of the resulting intervals with respect to the exact ones as the number of solutions increases. Although EMCC could potentially achieve a similar error with a large number of iterations, doing so would be extremely time-consuming. Thus, it is informative to analyze error against computation time, as shown in Figure 10. Here we can conclude that DCCC consistently approximates the bounds of counterfactual queries more accurately and in less time than EMCC. Note that some of the data-series in Figure 10 do not span the entire x-axis. This is because each experiment was terminated as soon as all possible solutions were visited (DCCC models with $|\mathrm{Pa}_Y| = 1$ and $\Omega_Y = 3$) or $N = 70$ solutions were found. Note also that we have truncated the x-axis in Figure 10 at 50 sec. instead of showing the full data-range (which is up to approximately 45 min. for the most time-consuming runs). This was done to make the behavior of the DCCC results clearly visible in the plots. Overall, the Gibbs sampling approach is more efficient than EMCC and only slightly slower than DCCC in generating each solution. However, since each new solution tends to be similar to the previous ones, the resulting low variability leads to a high error level unless a large number of solutions is generated.

Finally, to emphasise the potential of the presented heuristic search, some

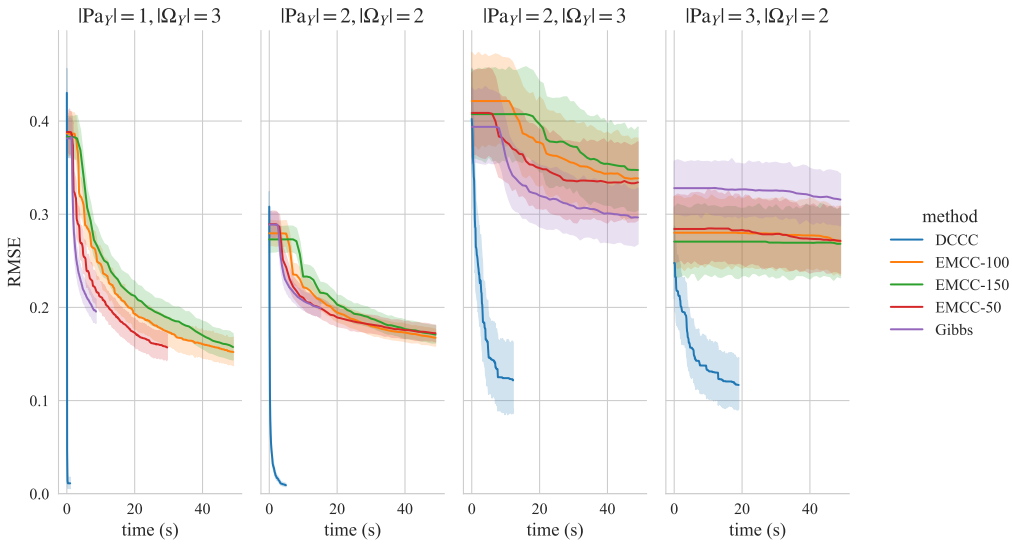


Figure 10: Error with respect the exact bounds vs. run time.

initial results are included showing the performance of DCCC on models of increased complexity, see Table 5. These results are restricted to models of two endogenous variables only, shown in the leftmost graph of Figure 6. The domain sizes of both endogenous variables (X_1 and Y) are varied for increased model complexity.

The counterfactual query $P(Y_{X_1=1} = 1|Y = 0, X_1 = 0)$ is chosen for these experiments, as it is a valid query for all domain sizes $|\Omega_Y| \geq 2$, $|\Omega_{X_1}| \geq 2$. Under the assumed simple model structure, the true interval may be calculated as

$$P(Y_{X_1=1} = 1|Y = 0, X_1 = 0) \in \left[1 - \frac{\min\{1 - P(Y = 1|X_1 = 1), P(Y = 0|X_1 = 0)\}}{P(Y = 0|X_1 = 0)}, \frac{\min\{P(Y = 1|X_1 = 1), P(Y = 0|X_1 = 0)\}}{P(Y = 0|X_1 = 0)}\right],$$

Domain sizes		Solution space			Results	
$\lvert\Omega_Y\rvert$	$\lvert\Omega_{X_1}\rvert$	$\lvert\Omega_{U_Y}\rvert$	Equations	$\binom{\lvert\Omega_{U_Y}\rvert}{(\lvert\Omega_Y\rvert-1)\cdot\lvert\Omega_{X_1}\rvert+1}$	Interval %	Time (s)
2	4	16	5	4368	100.00	0.06
2	6	64	7	$6.2 \cdot 10^8$	100.00	0.47
2	8	256	9	$1.1 \cdot 10^{16}$	98.63	15.95
2	10	1024	11	$3.1 \cdot 10^{25}$	69.47	60.00
2	12	4096	13	$1.4 \cdot 10^{37}$	8.73	60.00
3	4	81	9	$2.6 \cdot 10^{11}$	100.00	4.23
3	6	729	13	$2.4 \cdot 10^{27}$	19.43	57.86
4	2	16	7	11440	100.00	0.10
4	4	256	13	$2.4 \cdot 10^{21}$	55.16	55.49

Table 5: The table shows the result of interval approximations with DCCC for counterfactual query $P(Y_{X_1=1} = 1|Y = 0, X_1 = 0)$, for the leftmost model of Figure 6 with varied endogenous domain sizes, for randomly generated probability distributions $\tilde{P}(Y|X_1)$. The number of independent equations is given by $(|\Omega_Y| - 1) \cdot |\Omega_{X_1}| + 1)$. The two rightmost columns show the interval percentage retrieved and the average time of retrieval. For each domain size combination, the results are averages across 20 different $\tilde{P}(Y|X_1)$. The search is terminated once the complete interval is retrieved, or at 60 seconds. Thus, the average time shown in the result, if less than 60, suggest at least one complete interval is retrieved. If the interval percentage is 100, the average time is read as the average time to complete interval retrieval.

for reference when evaluating the approximations.

The results in Table 5 show the interval percentage retrieved in the first 60 seconds of running DCCC (with Apple M1 Max w/ 64 GB memory). The table details the number of variables (equal to $|\Omega_{U_Y}|$) and the number of independent equations of the linear system. For $|\Omega_Y| = 2, |\Omega_{X_1}| = 8$, this is a system of 9 equations in 256 variables, for which the results indicate that DCCC in most cases succeed in retrieving the complete interval within

the first minute of searching for solutions. Increasing the size of the domain of X_1 to 12, drastically increases the size of the system, which now has 13 equations in 4096 variables. Despite this huge number of variables, DCCC initiates interval approximation, and is seen to complete on average 8.73% of the interval in 60 seconds. Note that in this case, the total size of the search space is $\binom{4096}{13} \approx 10^{37}$. The heuristics employed, while less efficient, still succeeds in finding a subset of the extreme models. Here, DCCC benefits by not requiring explicit representation of the linear system to be solved. The search instead implicitly respects these equations by the base-$|\Omega_Y|$ numeral system function representation (Section 4.5). Only reduced linear systems of unique solutions are made explicit, which in the case of $|\Omega_Y| = 2, |\Omega_{X_1}| = 12$ is 13 equations with 13 unknowns.

Note that while further experiments would be required to determine time estimates for complete retrieval of query intervals as model complexity increases, all intervals retrieved are contained within the complete interval. Thus, while incomplete, they contain information about the true model.

The results in Table 5 assume canonical model specifications with maximum values for Ω_{U_Y}, and as such the results reflect a worst case performance for each domain size combination. The restriction to one endogenous parent is for the benefit of interval percentage calculations, as exact query bounds may be calculated. Approximating intervals for more general models is however expected to be of comparable performance, with a model with a binary Y having three binary parents X_1, X_2, X_3 corresponding in complexity to the $|\Omega_Y| = 2, |\Omega_{X_1}| = 8$ setting described above, etc.

6. Conclusions and future work

In this paper we propose a method for bounding unidentifiable queries in SCMs using a divide and conquer strategy to transform a general causal model into a set of models with low-cardinality exogenous variables, in which we can calculate any query using standard Bayesian network inference. Bounds for the query in the original model are then efficiently approximated by aggregating the results from these smaller models. We have shown in Section 4 that the DCCC method is technically sound, and the experiments reported in Section 5 validate the hypothesis that the proposed method is more efficient and accurate than current state of the art.

We envision at least two directions for future research: Firstly, we want to extend the applicability of the approach to more general model classes, like SCMs including hidden confounders (i.e. non-Markovian). Secondly, we intend to delve into novel methods for designing efficient heuristics to prioritize reductions. By incorporating query-specific knowledge into the reduction selection process, we anticipate significant performance gains.

Acknowledgments

This paper is an extended version of [10]. The most important extensions include the proofs of formal correctness for the approach as well as the extension beyond binary variables.

ARB and HL acknowledge the support by the Norwegian Research Council through grant 304843. RC and AS acknowledge Grant PID2022-139293NB-C31 funded by MICIU/AEI/10.13039/501100011033 and by ERDF "A way

of making Europe". RC and AS acknowledge the University of Almería Research and Transfer Programme funded by "Consejería de Universidad, Investigación e Innovación de la Junta de Andalucía" through the European Regional Development Fund (ERDF), Operation Programme 2021-2027. Programme: Research and Innovation 54.A. In brief: PPIT-UAL, Junta de Andalucía- ERDF 2021-2027. Programme: 54.A. RC was also supported by "Plan Propio de Investigación y Transferencia 2024-2025" from University of Almería under the project P_LANZ_2024/003.

Appendix A.

For running the experiments with the DCCC method presented in the paper, the model search is based on a CNF solution generator that generates partial solutions according to the heuristics detailed in Sections 4.4 and 4.5. The generator is implemented based on the following matrix structure describing a CNF solution:

$$\begin{pmatrix} a_{0,0} & a_{0,1} & \dots & a_{0,c-1} \\ a_{1,0} & a_{1,1} & \dots & a_{1,c-1} \\ \vdots & \vdots & & \vdots \\ a_{p-1,0} & a_{p-1,1} & \dots & a_{p-1,c-1} \end{pmatrix}$$

Here c is the CNF solution size, meaning the number of variables z_i included in solution set $\mathcal{Z}$ such that no variable may be removed with the remaining variables of $\mathcal{Z}$ still satisfying the formula. For a set of equations defining the credal set $\mathcal{K}(U)$ for exogenous U of model component $\mathbf{X} \to Y \leftarrow U$, with $|\Omega_{\mathbf{X}}| = p$, $|\Omega_Y| = q$ and $|\Omega_U| = q^p$, CNF solutions may be generated for any

c s.t. $q \leq c \leq p \cdot (q-1)$. Here, the minimum size at $c = q$ means the solution contains exactly one variable per clause (an example is highlighted in red color in Figure 5 for $c = q = 4$), while increasing c will increase the number of variables present in a subset of the clauses (an example where $c = 6$ is seen in blue in Figure 5). For any c, every variable z_i in the solution will be the only variable present in at least one of the clauses to ensure that the CNF solution is irreducible. Having $c > q$ is required for the probability-guided heuristic, where clauses for which a single z_i is part of the solution are assigned according to low observed probabilities, while clauses for which more than one z_i are part of the solution are assigned for higher observed probabilities.

To generate solutions of size c, let each column in the above defined matrix correspond to the base-q encoding of the index i of a variable to be included, thus for each element $a_{m,n}$, $0 \leq a_{m,n} \leq q-1$.

Each clause represents an equation in the set defining $\mathcal{K}(U)$, which in total has $q \cdot p$ equations. Each clause corresponds to a unique pair (j, k), with $0 \leq j \leq p-1$ and $0 \leq k \leq q-1$, such that clause (j, k) represents the equation for observed probability $p(Y = y_k | \mathbf{X} = \mathbf{x}_j)$.

A generated matrix solution should comply with the following restrictions:

1. Each row must include all digits in $\{0, 1, ..., q-1\}$ at least once. This ensures that the solution represented by the matrix satisfies the CNF formula. If a digit k is not present in a row j, then the clause corresponding to $p(Y = y_k | \mathbf{X} = \mathbf{x}_j)$ is not satisfied, and the complete formula will not be satisfied.

2. Each column must include at least one element that is unique in its re-

spective row. This ensures irreducibility of the corresponding solution. If a variable is removed such that the corresponding column is removed from the matrix, then at least one row is left incomplete according to restriction 1.

3. In order to apply the probability guided heuristic, the CNF solutions generated are restricted to contain exactly one variable appearing in clauses corresponding to the lowest probability clauses. Restricting a clause to exactly one variable corresponds to restricting the corresponding digit k to occur exactly once in row j.

The CNF solution generator generates solutions by considering permutations of the matrix structure under these restrictions, for any given size c. Next, CNF solutions are expanded to required size $p \cdot (q-1)+1$, where the simplest approach is to include variables at random from the remaining set. Finally, the z_i in the expanded solution are mapped to their corresponding $p(u_i)$ and the unique solution equation set is solved. If the solution to the equations set corresponds to a probability distribution, this distribution is returned as an extreme model M and included in the set used for query calculation.

References

[1] J. Pearl, Causality. Models, inference and reasoning. Second edition, Cambridge University Press, New York, 2009.

[2] E. Bareinboim, J. D. Correa, D. Ibeling, T. Icard, On Pearl's hierarchy

and the foundations of causal inference, in: Probabilistic and causal inference: the works of Judea Pearl, ACM, 2022, pp. 507–556.

[3] C. Kang, J. Tian, Inequality constraints in causal models with hidden variables, in: R. Dechter, T. Richardson (Eds.), Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence, AUAI Press, 2006, p. 233–240.

[4] M. C. Sachs, G. Jonzon, A. Sjölander, E. E. Gabriel, A general method for deriving tight symbolic bounds on causal effects., Journal of Computational and Graphical Statistics 32 (2) (2023) 567–576.

[5] J. Zhang, J. Tian, E. Bareinboim, Partial counterfactual identification from observational and experimental data, in: K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, S. Sabato (Eds.), Proceedings of the Thirty-Ninth International Conference on Machine Learning, Vol. 162 of ICML'22, JMLR.org, 2022, pp. 26548–26558.

[6] F. G. Cozman, Credal networks, Artificial intelligence 120 (2) (2000) 199–233.

[7] M. Zaffalon, A. Antonucci, R. Cabañas, Structural causal models are (solvable by) credal networks, in: International Conference on Probabilistic Graphical Models, PMLR, 2020, pp. 581–592.

[8] M. Zaffalon, A. Antonucci, R. Cabañas, D. Huber, D. Azzimonti, Efficient computation of counterfactual bounds, International Journal of Approximate Reasoning (2024) 109111.

[9] D. Koller, N. Friedman, Probabilistic graphical models: principles and techniques, MIT press, 2009.

[10] A. R. Bjøru, R. Cabañas, H. Langseth, A. Salmerón, A divide and conquer approach for solving structural causal models, in: J. Kwisthout, S. Renooij (Eds.), Proceedings of The 12th International Conference on Probabilistic Graphical Models, Vol. 246 of Proceedings of Machine Learning Research, 2024, pp. 348–360.

[11] S. Mueller, A. Li, J. Pearl, Causes of effects: Learning individual responses from population data, arXiv preprint arXiv:2104.13730 (2021).

[12] C. Avin, I. Shpitser, J. Pearl, Identifiability of path-specific effects, in: IJCAI International Joint Conference on Artificial Intelligence, 2005, pp. 357–363.

[13] J. Correa, S. Lee, E. Bareinboim, Nested counterfactual identification from arbitrary surrogate experiments, Advances in Neural Information Processing Systems 34 (2021) 6856–6867.

[14] R. Cabañas, A. D. Maldonado, M. Morales, P. A. Aguilera, A. Salmerón, Bayesian networks for causal analysis in socioecological systems, arXiv preprint arXiv:2401.10101 (2024).

[15] P. Walley, BI statistical methods. Volume 1: foundations, Prescience Press, 2015.