Journal of Statistical Software

August 2021, Volume 99, Issue 5. *doi: 10.18637/jss.v099.i05*

Causal Effect Identification from Multiple Incomplete Data Sources: A General Search-Based Approach

Santtu Tikka
University of Jyvaskyla

Antti Hyttinen
University of Helsinki

Juha Karvanen
University of Jyvaskyla

Abstract

Causal effect identification considers whether an interventional probability distribution can be uniquely determined without parametric assumptions from measured source distributions and structural knowledge on the generating system. While complete graphical criteria and procedures exist for many identification problems, there are still challenging but important extensions that have not been considered in the literature such as combined transportability and selection bias, or multiple sources of selection bias. To tackle these new settings, we present a search algorithm directly over the rules of do-calculus. Due to the generality of do-calculus, the search is capable of taking more advanced data-generating mechanisms into account along with an arbitrary type of both observational and experimental source distributions. The search is enhanced via a heuristic and search space reduction techniques. The approach, called do-search, is provably sound, and it is complete with respect to identifiability problems that have been shown to be completely characterized by do-calculus. When extended with additional rules, the search is capable of handling missing data problems as well. With the versatile search, we are able to approach new problems for which no other algorithmic solutions exist. We perform a systematic analysis of bivariate missing data problems and study causal inference under case-control design. We also present the R package **dosearch** that provides an interface for a C++ implementation of the search.

Keywords: causality, do-calculus, selection bias, transportability, missing data, case-control design, meta-analysis.

1. Introduction

In many fields of science, a primary interest is determining causal effects, that is, distributions $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$, where variables $\mathbf{Y}$ are observed, variables $\mathbf{X}$ are intervened upon (forced to values irrespective of their natural causes) and variables $\mathbf{Z}$ are conditioned on (Pearl 2009). In this paper, instead of placing various parametric restrictions based on background knowledge, we are interested in the question of identifiability: can the causal effect be uniquely determined from the distributions (data) we have and a graph representing our structural knowledge on the generating causal system.

In the most basic setting we are identifying causal effects from a single observational input distribution, corresponding to passively observed data. To solve such problems more generally than what is possible with the back-door adjustment (Spirtes, Glymour, and Scheines 1993; Pearl 2009; Greenland, Robins, and Pearl 1999), Pearl (1995) introduced *do-calculus*, a set of three rules that together with probability theory enable the manipulation of interventional distributions. Shpitser and Pearl (2006b) and Huang and Valtorta (2006b) showed that do-calculus is complete by presenting polynomial-time algorithms whose each step can be seen as a rule of do-calculus or as an operation based on basic probability theory. The algorithms have a high practical value because the rules of do-calculus do not by themselves provide an indication on the order in which the rules should be applied. The algorithms save us from manual application of do-calculus, which is a tedious task in all but the simplest problems.

Since then many extensions of the basic identifiability problem have appeared. In identifiability using surrogate experiments (Bareinboim and Pearl 2012a), or z-identifiability, an experimental distribution is available in addition to the observed probability distribution. For data observed in the presence of selection bias, both algorithmic and graphical identifiability results have been derived (Bareinboim and Tian 2015; Correa, Tian, and Bareinboim 2018). More generally, the presence of missing data necessitates the representation of the missingness mechanism, which poses additional challenges (Mohan, Pearl, and Tian 2013; Shpitser, Mohan, and Pearl 2015; Bhattacharya, Nabi, Shpitser, and Robins 2019). Another dimension of complexity is the number of available data sources. Identification from a mixture of observational and interventional distributions that originate from multiple conceptual domains is known as transportability for which complete solutions exist in a specific setting (Bareinboim and Pearl 2014).

While completeness has been accomplished for a number of basic identifiability problems, there are still many challenging but important extensions to the identifiability problem that have not been studied so far. Table 1 recaps the current state of the art identifiability results; it also describes generalizations that we aim to investigate in this paper. To find solutions to the more complicated identifiability problems, we present a unified approach to the identification of observational and interventional causal queries by constructing a search algorithm that directly applies the rules of do-calculus. We impose no restrictions on the number or type of known input distributions: we thus provide a solution to problems for which no other algorithmic solutions exist (Row 8 in Table 1). We also extend to identifiability under missing data together with mechanisms related to selection bias and transportability (Row 11 in Table 1).

The following introductory example does not fall under any of the previously solved special cases of Table 1, thus necessitating our approach. Consider human resource management analyzing the remuneration policy in a company. The graph of Figure 1 shows the key variables. The salary (Y) of an employee consist of a base salary (not modeled explicitly) and a bonus (B). The base salary depends on education (E) and performance (X) of the employee. The level of performance is evaluated by the supervisor of the employee and it is one of the factors that affects the level of bonus. The performance depends on the education

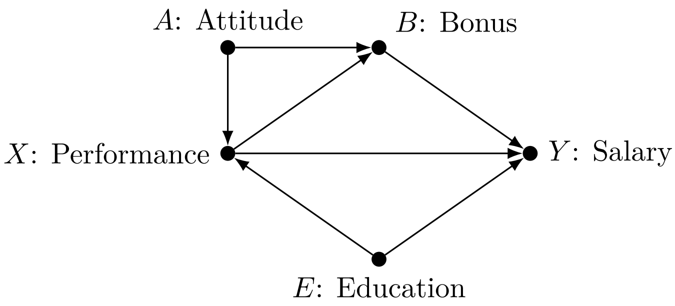


Figure 1: Graph for the example on human resource management in a company.

as well as the attitude (A) of the employee and the team. The attitude may also have a direct effect on the bonus.

We are interested in estimating the total causal effect of performance on the salary, i.e., quantifying how changes in performance affect the salary. The salary (Y), bonus (B), education (E) and performance (X) of the employees are available from the registry of the human resource department. Data on attitude (A), bonus (B) and performance (X) have been collected in an anonymized survey that cannot be linked to the registry on the personal level. The question of interest is the identifiability of $P(Y \mid \mathrm{do}(X))$ from the data sources $P(Y, B, E, X)$ and $P(A, B, X)$. Using the machinery developed in this paper, we can identify the causal effect with the formula

$$P(Y \mid \mathrm{do}(X)) = \sum_{B,A} P(A)P(B \mid X, A) \sum_{E} P(E)P(Y \mid X, B, E),$$

where the conditional distributions can be directly determined from the available data sources (see Section 6 for further details).

To combat the inherent computational complexity of the search-based approach, we derive rules and techniques that avoid unnecessary computational steps. We are able to detect trivial queries where non-identifiability can be determined directly from the inputs. We also present a search heuristic that considerably speeds up the search in cases where the effect is indeed identifiable. The approach, called do-search, is provably sound and it retains the completeness in the cases previously proven to be solved by the rules of do-calculus. We can easily scale up to the problem sizes commonly reported in the literature. The R package **dosearch** (R Core Team 2021; Tikka, Hyttinen, and Karvanen 2021) provides an implementation of do-search and is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=dosearch`.

Other available software for causal effect identifiability problems are only applicable to a subset of the problems presented in Table 1. The **causaleffect** R package (Tikka and Karvanen 2017a) can only be used for problems on Rows 1–3 and 5–7 of the table, providing implementations of the relevant algorithms. For the problem on Row 1, the generalized adjustment criterion (Perković, Textor, Kalisch, and Maathuis 2015) can be applied using the **dagitty** package (Textor, Van der Zander, Gilthorpe, Liśkiewicz, and Ellison 2016) or the **pcalg** package (Kalisch, Mächler, Colombo, Maathuis, and Bühlmann 2012) in R. The generalized back-door criterion (Maathuis and Colombo 2015) is also available in the **pcalg** R package. The standard back-door criterion is available in the Python package **DoWhy** (Sharma and

	Problem (Reference)	Target	Input (assumptions)	Missing data pattern	Method (complete)
1	Causal effect identifiability (Shpitser and Pearl 2006b)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}))$	$P(\mathbf{V})$	None	ID (Yes)
2	Causal effect identifiability (Shpitser and Pearl 2006a)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$P(\mathbf{V})$	None	IDC (Yes)
3	z-identifiability (Bareinboim and Pearl 2012a)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$P(\mathbf{V})$, $P(\mathbf{V} \setminus \mathbf{B} \mid \mathrm{do}(\mathbf{B}))$ (NE, ED)	None	zID (Yes)
4	g-identifiability (Lee, Correa, and Bareinboim 2019)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}))$	$\{P(\mathbf{V} \setminus \mathbf{B}_i \mid \mathrm{do}(\mathbf{B}_i)\}$ (ED)	None	gID (Yes)
5	Surrogate outcome identifiability (Tikka and Karvanen 2019)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$\{P(\mathbf{A}_i \mid \mathrm{do}(\mathbf{B}_i), \mathbf{C}_i)\}$ (NE, SO)	None	TRSO (No)
6	mz-transportability (Bareinboim and Pearl 2014)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$\{P(\mathbf{V} \setminus (\mathbf{B}_i \cup \mathbf{T}_i) \mid \mathrm{do}(\mathbf{B}_i), \mathbf{T}_i)\}$ (NEDD, ED)	None	$\mathrm{TR}^{\mathrm{mz}}$ (Yes)
7	Selection bias recoverability (Bareinboim and Tian 2015)	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$P(\mathbf{V} \setminus S \mid S)$	Selection	RC (Unknown)
8	***Generalized identifiability***	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$\{P(\mathbf{A}_i \mid \mathrm{do}(\mathbf{B}_i), \mathbf{C}_i)\}$	***None***	***do-search*** (Unknown)
9	Missing data recoverability (Mohan *et al.* 2013)	$P(\mathbf{V})$	$P(\mathbf{V}^*)$	Restricted	– (Yes)
10	Missing data recoverability (Bhattacharya *et al.* 2019)	$P(\mathbf{V})$	$P(\mathbf{V}^*)$	Arbitrary	– (Unknown)
11	***Generalized identifiability with missing data***	$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z})$	$\{P(\mathbf{A}_i^* \mid \mathrm{do}(\mathbf{B}_i), \mathbf{C}_i^*)\}$	***Arbitrary***	***do-search*** (No)

Table 1: Solved and unsolved problems in causal effect identification. Bold-italic denotes the previously unsolved problems for which do-search can now be used. Input $P(\mathbf{V})$ stands for passively observed joint distribution of all variables. Input $P(\mathbf{V}^*)$ is the joint distribution with missing data (see Section 4). The variable sets present in the same distribution are disjoint. Input $P(\mathbf{V} \setminus \mathbf{B} \mid \mathrm{do}(\mathbf{B}))$ stands for an experiment where all variables are measured and input $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}))$ stands for an experiment where only a subset $\mathbf{A} \subset \mathbf{V}$ of the variables is measured. Notation $\{\cdot\}$ denotes a set of inputs enumerated by the index i. The assumptions of nested experiments (NE), entire distributions (ED) and nested experiments in different domains (NEDD) are explained in Section 2. Assumptions related to surrogate outcomes (SO) can be found in (Tikka and Karvanen 2019). Input $P(\mathbf{V} \mid S)$ means the joint distribution under selection bias. The last column gives the name of an algorithm that can be used to solve the problem if one exists and whether it (or a theorem when no algorithm is provided) provides a complete solution to the problem, or whether the completeness status is not known. An algorithm is complete if it returns a correct formula precisely when the target query is identifiable. Problems 1–7 are special cases of Problem 8 and Problems 1–10 are special cases of Problem 11.

Kiciman 2019). Algorithms implemented in **causaleffect** run in polynomial time and can outperform **dosearch** in their respective *restricted* problem settings especially with larger graphs. For a comprehensive performance comparison between **causaleffect** and various adjustment criteria, see (Van der Zander, Liśkiewicz, and Textor 2019).

The paper is structured as follows. Section 2 formulates our general identification problem and explains the scenarios in Table 1 and previous research in detail. Section 3 presents the search algorithm, including the rules we use, search space reduction techniques, heuristics and theoretical properties. Section 4 shows how the search can be extended to problems that involve missing data. Section 5 demonstrates how the search can be used in R via the **dosearch** package. Efficacy of the search is assessed via simulations. Section 6 shows a number of new problems for which we can find solutions by using the search including a real-world application. These problems include combined transportability and selection bias, multiple sources of selection bias, and causal effect identification from arbitrary (experimental) distributions. This section also includes a systematic analysis of missing data problems and case-control designs. Section 7 discusses the merits and limitations of the approach. Section 8 offers concluding remarks.

2. The general causal effect identification problem

Our presentation is based on structural causal models (SCM) and the language of directed graphs. We assume the reader to be familiar with these concepts and refer them to detailed works on these topics for extended discussion and descriptions, such as (Pearl 2009) and (Koller and Friedman 2009).

Following the standard set-up of do-calculus (Pearl 1995), we assume that the causal structure can be represented by a *semi-Markovian causal graph* G over a set of vertices $\mathbf{V}$ (see Figure 2(a) for example). The directed edges correspond to direct causal relations between the variables (relative to $\mathbf{V}$); directed edges do not form any cycles. Confounding of any two observed variables in $\mathbf{V}$ by some unobserved common cause is represented by a bidirected edge between the variables. This graphical representation allows us to deal with any causal structures where some variables are unmeasured. We assume a positive distribution over the variables (Huang and Valtorta 2006a) ensuring that all considered causal effects and conditional distributions are well-defined.

In a non-parametric setting, the problem of expressing a causal quantity of interest in terms of available information has been described in various ways depending on the context. When available data are affected by selection bias or missing data, a typical goal is to "recover" a joint or marginal distribution. If data are available from multiple conceptual domains, a distribution is "transported" from the source domains, from which a combination of both observational and experimental data are available, to a target domain. The aforementioned settings can be expressed in the SCM framework by equipping the graph of the model with special vertices. However, on a fundamental level these problems are simply variations of the original identifiability problem of causal effects and as such, our goal is to represent them as a single generalized identifiability problem. Formally, identifiability can be defined as follows (Pearl 2009; Shpitser and Pearl 2008).

Definition 1 (Identifiability)**.** *Let* $\mathbf{M}$ *be a set of models with a description* T *and two objects* ϕ *and* θ *computable from each model. Then* ϕ *is* identifiable *from* θ *in* T *if* ϕ *is uniquely computable from* θ *in any model* $M \in \mathbf{M}$*. In other words, all models in* $\mathbf{M}$ *which agree on* θ *also agree on* ϕ*.*

In the simplest case, the description T refers to the graph induced by causal model, θ is the joint distribution of the observed variables $P(\mathbf{V})$ and the query ϕ is a causal effect

$P(Y \mid \text{do}(X))$. On the other hand, proving non-identifiability of ϕ from θ can be obtained by describing two models $M^1, M^2 \in \mathbf{M}$ such that θ is the same in M^1 and M^2, but object ϕ in M^1 is different from ϕ in M^2.

The general form for a causal identifiability problem that we consider in this paper is formulated as follows.

Input: A set of input distributions of the form $P(\mathbf{A}_i \mid \text{do}(\mathbf{B}_i), \mathbf{C}_i)$, a query $P(\mathbf{Y} \mid \text{do}(\mathbf{X}), \mathbf{Z})$ and a semi-Markovian causal graph G over $\mathbf{V}$.

Task: Output a formula for the query $P(\mathbf{Y} \mid \text{do}(\mathbf{X}), \mathbf{Z})$ over the input distributions, or decide that it is not identifiable.

Here $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ are disjoint subsets of $\mathbf{V}$ for all i, and $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ are disjoint subsets of $\mathbf{V}$. The causal graph G may contain vertices which describe mechanisms related to transportability and selection bias. In the following sections we explain several important special cases of this problem definition, some that have been considered in the literature and some which have not been.

2.1. Previously considered scenarios as special cases

We restate the concepts of transportability and selection bias under the causal inference framework, and show that identifiability in the scenarios of Rows 1–7 of Table 1 falls under the general form on Row 8. We return to problems that involve missing data on Rows 9–11 later in Section 4.

Causal effect identifiability

Input is restricted to a passive observational distribution $P(\mathbf{V})$. The target is either a causal effect $P(\mathbf{Y} \mid \text{do}(\mathbf{X}))$ for Row 1 of Table 1 or a conditional causal effect $P(\mathbf{Y} \mid \text{do}(\mathbf{X}), \mathbf{Z})$ for Row 2 of Table 1 (Shpitser and Pearl 2006b,a).

z-identifiability

Similarly to ordinary causal effect identification, the input consists of the passive observational distribution $P(\mathbf{V})$ but also of experimental distributions known as surrogate experiments intervening on a set $\mathbf{B}$ (Bareinboim and Pearl 2012a). Two restricting assumptions, which we call nested experiments and entire distributions, apply to surrogate experiments. Experiments are called nested experiments (NE) when for each experiment intervening a set of variables $\mathbf{B}$, experiments intervening on all subsets of $\mathbf{B}$ are available as well. Entire distributions (ED) denote the assumption that the union of observed and intervened variables is always the set of all variables $\mathbf{V}$.

g-identifiability

Unlike z-identifiability, the input does not contain the passive observational distribution $P(\mathbf{V})$ but consists instead of surrogate experiments on the sets $\{\mathbf{B}_i\}$ (Lee *et al.* 2019) without the assumption of nested experiments. The assumption of entire distributions holds.

Surrogate outcome identifiability

Surrogate outcomes generalize the notion of surrogate experiments from z-identifiability. For surrogate outcomes, the assumption of nested experiments still holds, but the assumption of entire distributions can be dropped. Some less strict assumptions (SO) still apply (Tikka and Karvanen 2019). The idea of surrogate outcomes is that data from previous experiments are available, but the target $\mathbf{Y}$ was at most only partially measured in these experiments and the experiments do not have to be disjoint from $\mathbf{X}$.

Transportability

The problem of incorporating data from multiple causal domains is known as transportability (Bareinboim and Pearl 2013). Formally, the goal is to identify a query in a target domain π^* using data from source domains $\pi_1, \ldots, \pi_n$. The domains are represented in the causal graph using a special set of transportability nodes $\mathbf{T}$ which is partitioned into disjoint subsets $\mathbf{T}_1, \ldots, \mathbf{T}_n$ corresponding to each domain π_i. The causal graph contains an extra edge $T_{ij} \rightarrow V_j$ whenever a functional discrepancy in f_{V_j} or in $P(u_{V_j})$ exists between the target domain π^* and the source domain π_i. The discrepancy is active if $T_{ij} = 1$ and inactive otherwise. A distribution associated with a domain π_i is of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C}, \mathbf{T}_i = 1, \mathbf{T}_{-i} = 0)$, where $\mathbf{T}_{-i}$ denotes the other subsets of the partition of $\mathbf{T}$ except $\mathbf{T}_i$. In other words, only the discrepancies between the π_i and π^* are active. A distribution corresponding to the target domain has no active discrepancies meaning that it is of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C}, \mathbf{T} = 0)$. Any variable is conditionally independent from inactive transportability nodes since their respective edges vanish. Furthermore, since transportability nodes set to 0 vanish, we can assume any present transportability node to have the value 1. Thus an input distribution from a domain π_i takes the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C}, \mathbf{T}_i)$. In the specific case of mz-transportability, the assumptions of entire distributions (ED) and nested experiments in different domains (NEDD) apply, which means that $P(\mathbf{V} \setminus (\mathbf{B}'_i \cup \mathbf{T}_i) \mid \mathrm{do}(\mathbf{B}'_i), \mathbf{T}_i)$ is available for every subset $\mathbf{B}'_i$ of $\mathbf{B}_i$ in each domain π_i.

Selection bias recoverability

Selection bias can be seen as a special case of missing data, where the mechanism responsible for the preferential selection is represented in the causal graph by a special sink vertex S (Bareinboim and Pearl 2012b). Typical input for the recoverability problem is $P(\mathbf{V} \mid S = 1)$, the joint distribution observed under selection bias. Just as in the case of transportability nodes, selection bias nodes only appear when the mechanism has been enabled. Thus we may assume that the input is of form $P(\mathbf{V} \mid S)$. More generally, we can consider input distributions of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C}, S)$.

2.2. New scenarios as special cases

The following settings are special cases of the general identifiability problem of Row 8 in Table 1 that do not fall under any of the problems of Rows 1–7. They serve as interesting additions to the cases considered in the literature. Concrete examples on these new scenarios are presented in Section 6. Section 4 extends the general problem of Row 8 in Table 1 to the general problem with missing data on Row 11 while also showcasing the special cases of Rows 9 and 10.

Multiple data sources with partially overlapping variable sets

The scenario where only subsets of variables are ever observed together has been extensively considered in the causal discovery literature (Danks, Glymour, and Tillman 2009; Tillman and Spirtes 2011; Triantafillou, Tsamardinos, and Tollis 2010), but not in the context of causal effect identification. In the basic setting the input consists of passively observed distributions $P(\mathbf{A}_i)$ such that $\mathbf{A}_i \subset \mathbf{V}$. We may also observe experimental distributions $P(\mathbf{A}_i \mid \mathrm{do}(\mathbf{B}_i))$ (Hyttinen, Eberhardt, and Hoyer 2012; Triantafillou and Tsamardinos 2015) or even conditionals $P(\mathbf{A}_i \mid \mathrm{do}(\mathbf{B}_i), \mathbf{C}_i)$. Our approach sets no limitations for the number or types of input distributions.

Combining transportability and selection bias

To the best of our knowledge, the frameworks of transportability and selection bias have not been considered simultaneously. The combination of these scenarios fits into the general problem formulation. For example, we may have access to two observational distributions originating from different source domains, but affected by the same biasing mechanism: $P(\mathbf{A}_1 \mid \mathbf{C}_1, T_1, S)$ and $P(\mathbf{A}_2 \mid \mathbf{C}_2, T_2, S)$, where T_1 and T_2 are the transportability nodes corresponding to the two source domains and S is the selection bias node.

Recovering from multiple sources of selection bias

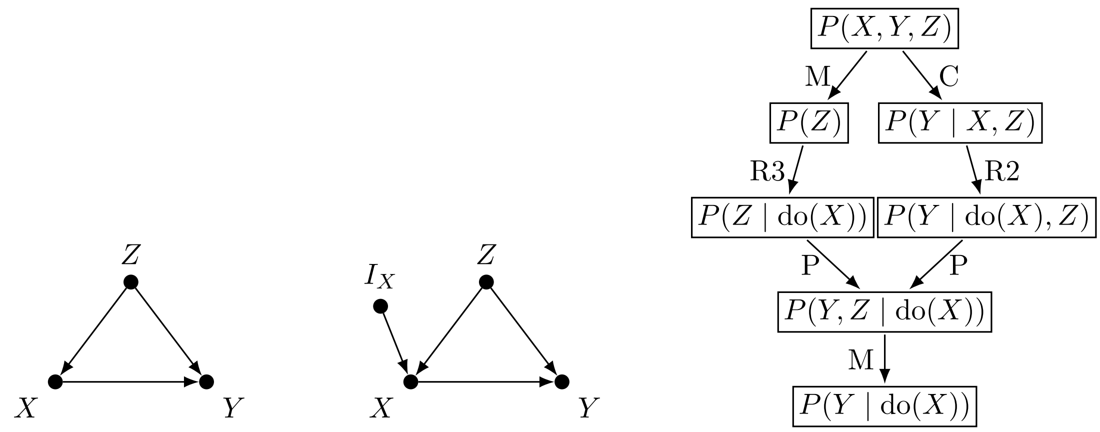
In recent literature on selection bias as a causal inference problem, the focus has been on settings where only a single selection bias node is present (e.g., Bareinboim, Tian, and Pearl 2014; Correa and Bareinboim 2017; Correa *et al.* 2018). However, multiple sources of selection bias are typical in longitudinal studies where dropout occurs at different stages of the study. Our approach is applicable for an arbitrary number of selection bias mechanisms and input distributions affected by arbitrary combinations of these mechanisms. In other words, if $\mathbf{S}$ is the set of all selection bias nodes present in the graph, the inputs can take the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C}, \mathbf{S}')$, where $\mathbf{S}'$ is an arbitrary subset of $\mathbf{S}$.

3. A search-based approach for causal effect identification

The key to identification of causal effects is that interventional expressions can be manipulated using the rules of do-calculus. We present these rules for augmented graphs where an additional intervention variable I_X such that $I_X \rightarrow X$ is added to the induced graph for each variable X (Spirtes *et al.* 1993; Pearl 2009; Lauritzen 2000) (see Figure 2(b)). Now a d-separation condition (or m-separation (Richardson 2003)) of the form $\mathbf{Y} \perp\!\!\!\perp \mathbf{I_Z} \mid \mathbf{X}, \mathbf{Z}, \mathbf{W} \,||\, \mathbf{X}$ means that nodes $\mathbf{Y}$ and intervention nodes $\mathbf{I_Z}$ of $\mathbf{Z}$ are d-separated given $\mathbf{X}$, $\mathbf{Z}$ and $\mathbf{W}$ in a graph where edges incoming to (intervened) $\mathbf{X}$ have been removed (Hyttinen, Eberhardt, and Järvisalo 2015; Dawid 2002). The three rules of do-calculus (Pearl 1995) can be expressed as follows:

$$
\begin{aligned}
P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W}) &= P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}), \text{ if } \mathbf{Y} \perp\!\!\!\perp \mathbf{Z} \mid \mathbf{X}, \mathbf{W} \,||\, \mathbf{X} \\
P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W}) &= P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W}), \text{ if } \mathbf{Y} \perp\!\!\!\perp \mathbf{I_Z} \mid \mathbf{X}, \mathbf{Z}, \mathbf{W} \,||\, \mathbf{X} \\
P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W}) &= P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}), \text{ if } \mathbf{Y} \perp\!\!\!\perp \mathbf{I_Z} \mid \mathbf{X}, \mathbf{W} \,||\, \mathbf{X}
\end{aligned}
\qquad (1)
$$

The rules are often referred to as insertion/deletion of observations, exchange of actions and observations, and insertion/deletion of actions respectively. Each rule of do-calculus is



(a) Example graph. (b) Augmented graph. (c) A derivation for $P(Y \mid \mathrm{do}(X))$.

Figure 2: The back-door criterion holds in the example graph (a) for Z. The augmented graph (b) includes the intervention node I_X for X explicitly. The labels M, C, P, R2 and R3 in the derivation of (c) refer to marginalization, conditioning, product rule and Rules 2 and 3 of do-calculus respectively (see Table 2). The required d-separation conditions $Y \perp\!\!\!\perp I_X \mid Z, X$ for R2 and $Z \perp\!\!\!\perp I_X$ for R3 hold in the augmented graph (b).

Algorithm 1 An outline of a search for causal effect identification.

Input: Target $Q = P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$, a semi-Markovian graph G and a set of known input distributions $\mathbf{P} = \{P_1, \ldots, P_n\}$.

Output: A formula for Q or NA if the effect is not identifiable.

for each $P_i \in \mathbf{P}$ **do**
    Derive new distributions from P_i such that:
    - The required d-separation criteria are satisfied by G.
    - Any possible additional input required must also be in $\mathbf{P}$.

    Add the new identified distributions to $\mathbf{P}$.
    If Q was derived, return a formula for it.
Return NA.

only applicable if the accompanying d-separation criterion (on the right-hand side) holds in the underlying graph. In addition to these rules, most derivations require basic probability calculus. Do-calculus directly motivates a forwards search over its rules. The outline of this type of search is given in Algorithm 1. The algorithm derives new identifiable distributions based on what has been given as the input or identified in the previous steps. For each identified distribution every rule of do-calculus and standard probability manipulations of marginalization and conditioning are applied in succession, until the target distribution is found, or no new distributions can be found to be identifiable. A preliminary version of this kind of search is used by Hyttinen *et al.* (2015) as a part of an algorithmic solution to causal effect identifiability when the underlying graph is unavailable.

The formulas produced by Algorithm 1 correspond to short derivations and unnecessarily complicated expressions are avoided. Also, only distributions guaranteed to be identifiable are derived and used during the search. Formulas for intermediary queries that were identified during the search are also available as a result. Alternatively, one could also start with

Rule	Additional Input	Output	Description
1+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W})$	Insertion of observations
1−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	Deletion of observations
2+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W} \setminus \mathbf{Z})$	Observation to action exchange
2−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X} \setminus \mathbf{Z}), \mathbf{Z}, \mathbf{W})$	Action to observation exchange
3+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W})$	Insertion of actions
3−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X} \setminus \mathbf{Z}), \mathbf{W})$	Deletion of actions
4		$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Marginalization
5		$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W})$	Conditioning
6+	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	Chain rule multiplication
6−	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Y}, \mathbf{W})$	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Chain rule multiplication

Table 2: The rules used to manipulate input distributions of the form $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$. The output distribution is identified if the input has been previously identified and if the corresponding d-separation Criteria 1 hold in the graph (for Rules $1\pm, 2\pm$ and $3\pm$) or if the additional input has also been identified (Rules $6\pm$). The sets $\mathbf{Y}, \mathbf{X}$ and $\mathbf{W}$ are disjoint. The role of the set $\mathbf{Z}$ depends on the rule being applied (see Table 3).

the target and search towards the input distributions; a search in this direction will spend time deriving a number expressions that are inevitably non-identifiable based on the input. A depth-first search would produce unnecessarily complicated expressions. The search can easily derive for example the back-door criterion in the graph of Figure 2(a) as shown by the derivation in Figure 2(c). The target is $Q = P(Y \mid \mathrm{do}(X))$ and input is $\mathbf{P} = \{P(X, Y, Z)\}$. From $P(X, Y, Z)$ the search first derives the marginal $P(Z)$ and the conditional $P(Y \mid X, Z)$. Then $P(Z \mid \mathrm{do}(X))$ is derived by the third rule of do-calculus because $Z \perp\!\!\!\perp I_X$. The second rule derives $P(Y \mid \mathrm{do}(X), Z)$ from $P(Y \mid X, Z)$ as $Y \perp\!\!\!\perp I_X \mid Z, X$. The two terms can be combined via the product rule of probability calculus to get $P(Y, Z \mid \mathrm{do}(X))$ and finally the target is $P(Y \mid \mathrm{do}(X))$ is just a marginalization of this. The familiar formula $\sum_Z P(Y \mid X, Z)P(Z)$ is thus obtained.

However, it is not straightforward to make a search over do-calculus computationally feasible. The search space in Figure 2(c) shows only the parts that resulted in the identifying formula: for example all passively observed marginals and conditionals over $\mathbf{V}$ can be derived from the input $P(\mathbf{V})$. Especially in a non-identifiable case a naive search may go through a huge space before it can return the non-identifiable verdict. The choice of rules is also not obvious: a redundant rule may make the search faster or slower; false non-identifiability may be concluded if a necessary rule is missing. Also the order in which the rules are applied can have a large impact on the performance of the search. In the following sections we will provide non-trivial solutions to these challenges.

3.1. Rules

Table 2 lists the full set of rules used to manipulate distributions during the search, generalizing the work by Hyttinen *et al.* (2015).

Do-calculus

Rules $1\pm, 2\pm$ and $3\pm$ correspond to the rules of do-calculus such that Rules 1+, 2+ and 3+ are used to add conditional variables and interventions and Rules $1-, 2-, 3-$ are used to remove them. Each rule is only valid if the corresponding d-separation criterion given in the beginning of Section 3 holds.

Probability theory

Rule 4 performs marginalization over $\mathbf{Z} \subset \mathbf{Y}$, and produces a summation at the formula level:

$$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}) = \sum_{\mathbf{Z}} P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}).$$

Similarly, Rule 5 conditions on a subset $\mathbf{Z} \subset \mathbf{Y}$ to obtain the following formula:

$$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W}) = \frac{P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})}{\sum_{\mathbf{Y} \setminus \mathbf{Z}} P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})}.$$

Rules 6+ and 6− perform multiplication using the chain rule of probability which requires two known distributions. When Rule 6+ is applied, the distribution $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$ is known and we check whether $P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$ is known as well. For Rule 6−, the roles of the distributions are reversed. In the case of Rule 6+, $\mathbf{Z}$ is a subset of $\mathbf{W}$ and we obtain

$$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z}) = P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}) P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z}).$$

The two version of the chain rule are needed: it may be the case that when expanding $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$ with Rule 6+ the additional input $P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$ is only identified later in the search. Then, $P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$ is identified when Rule 6− is applied to $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$.

3.2. Improving the efficacy of the search

In this section, we present various techniques that improved the efficiency of the search. These findings are implemented in a search algorithm in Section 3.3.

Term expansion

Term expansion refers to the process of deriving new distributions from an input distribution using the rules of Table 2. By *term* we mean a single identified distribution. A term is considered *expanded* if the rules of Table 2 have been applied to it in every possible way when the term is in the role of the input. Note that an expanded distribution may still take the role of an additional input when another term is being expanded. Consider the step of expanding the input term in Table 2 to all possible outputs with any rule. This can be done by enumerating every non-empty subset $\mathbf{Z}$ of $\mathbf{V}$, and applying the rule with regard to it. Table 3 outlines the requirements for $\mathbf{Z}$ for each rule of the search. Table 3 tells us that when an observation $\mathbf{Z}$ is added using Rule 1+, it cannot be contained in any of the sets $\mathbf{Y}, \mathbf{X}$ or $\mathbf{W}$ since they are already present in the term. Only observations that are present can be removed, which is why $\mathbf{Z}$ has to a subset of $\mathbf{W}$ when applying Rule 1−. We may skip the application of this rule if the set of observations is empty for the current term. The exchange of observations to experiments using Rule 2+ has similar requirements for set $\mathbf{Z}$ as Rule 1−. Exchanging experiments to observations using Rule 2− works in a similar fashion. Only experiments that are present can be exchanged which means that $\mathbf{Z} \subseteq \mathbf{X}$. This rule can be skipped if the set of experiments is empty. New experiments are added using Rule 3+ with

Rule	Validity condition	Termination condition
1+	$\mathbf{Z} \cap (\mathbf{Y} \cup \mathbf{X} \cup \mathbf{W}) = \emptyset$	
1−	$\mathbf{Z} \subseteq \mathbf{W}$	$\mathbf{W} = \emptyset$
2+	$\mathbf{Z} \subseteq \mathbf{W}$	$\mathbf{W} = \emptyset$
2−	$\mathbf{Z} \subseteq \mathbf{X}$	$\mathbf{X} = \emptyset$
3+	$\mathbf{Z} \cap (\mathbf{Y} \cup \mathbf{X} \cup \mathbf{W}) = \emptyset$	
3−	$\mathbf{Z} \subseteq \mathbf{X}$	$\mathbf{X} = \emptyset$
4	$\mathbf{Z} \subset \mathbf{Y}$	$\lvert\mathbf{Y}\rvert = 1$
5	$\mathbf{Z} \subset \mathbf{Y}$	$\lvert\mathbf{Y}\rvert = 1$
6+	$\mathbf{Z} \subseteq \mathbf{W}$	$\mathbf{W} = \emptyset$
6−	$\mathbf{Z} \cap (\mathbf{Y} \cup \mathbf{X} \cup \mathbf{W}) = \emptyset$	

Table 3: The conditions for the enumerated subset $\mathbf{Z}$ for applying the rules of Table 2 to a term $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$. For Rules 6+ and 6−, the conditions specify valid variables of the second required term.

similar requirements as Rule 1+. Well-defined subsets for using Rule 3− are the same as for rule 2−. For Rules 4 and 5, the only requirement is that $\mathbf{Z}$ is a proper subset of $\mathbf{Y}$. When the chain rule is applied with Rule 6+, we require that the variables of the second product term is observed in the first term. When applied in reverse with Rule 6−, the variables of the second term must not be present in the first term.

Termination conditions

Additionally, Table 3 lists the termination condition for each rule: if it is satisfied by the current term to be expanded we know that the rule cannot be applied to it. The following simple lemma shows that when any of the termination conditions hold, no new distributions can be derived from it using the respective rule, which allows the search to directly proceed to the next rule.

Lemma 1. *Let G be a semi-Markovian graph and let $\mathbf{Y}, \mathbf{X}$ and $\mathbf{W}$ be disjoint subsets of $\mathbf{V}$. Then all of the following are true:*

(i) If $\mathbf{W} = \emptyset$, then Rule 1− of Table 2 cannot be used.

(ii) If $\mathbf{W} = \emptyset$, then Rule 2+ of Table 2 cannot be used.

(iii) If $\mathbf{X} = \emptyset$, then Rule 2− of Table 2 cannot be used.

(iv) If $\mathbf{X} = \emptyset$, then Rule 3− of Table 2 cannot be used.

(v) If $\lvert\mathbf{Y}\rvert = 1$, then Rule 4 of Table 2 cannot be used.

(vi) If $\lvert\mathbf{Y}\rvert = 1$, then Rule 5 of Table 2 cannot be used.

(vii) If $\mathbf{W} = \emptyset$, then Rule 6+ of Table 2 cannot be used.

Proof. For (i), the set $\mathbf{W}$ is empty so the application of Rule 1− using any subset $\mathbf{Z}$ would result in $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z}) = P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$ which is already identified. For (ii), the set $\mathbf{W}$ is empty so no observation can be exchanged for an action using the second rule of do-calculus. For (iii), the set $\mathbf{X}$ is empty so no action can be exchanged for an observation

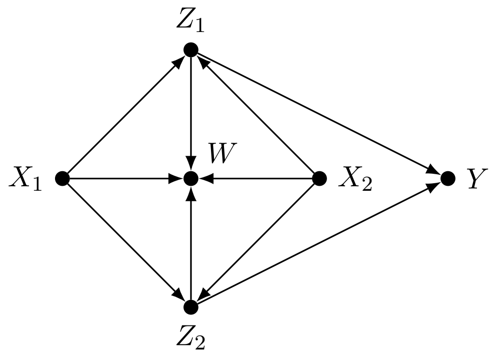


Figure 3: A graph for the example where all rules of Table 2 are required for identifying the target quantity.

using the second rule of do-calculus. For (iv), the set $\mathbf{X}$ is empty so the application of Rule 3− using any subset $\mathbf{Z}$ would result in $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X} \setminus \mathbf{Z}), \mathbf{W}) = P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$ which is already identified. For (v) and (vi), the set $\mathbf{Y}$ only has a single vertex, so it cannot have a non-empty subset. For (vii), the set $\mathbf{W}$ is empty so no subset $\mathbf{Z} \subset \mathbf{W}$ can exist for the second input. □

Rule necessity

The Rule 1 of do-calculus can be omitted as shown by Huang and Valtorta (2006b, Lemma 4). Instead of inserting an observation using Rule 1, we can insert an intervention and then exchange it for an observation. Similarly, an observation can be removed by first exchanging it for an intervention and then deleting the intervention. It follows that Rules 1+ and 1− of Table 2 are unnecessary for the search. The following example shows that the remaining rules of Table 2 are all necessary. In the graph of Figure 3, the causal effect $P(Y, X_1 \mid \mathrm{do}(X_2), W)$ can be identified from the inputs $P(W \mid \mathrm{do}(X_2), Y, X_1)$, $P(Y \mid \mathrm{do}(X_2), Z_1, Z_2, X_1)$, $P(X_1 \mid \mathrm{do}(X_2), W)$, $P(Z_2, X_2 \mid \mathrm{do}(X_1))$ and $P(Z_1 \mid \mathrm{do}(X_1, Y), X_2)$ when all rules are available, but not when any individual rule is omitted. This can be verified by running the search algorithm presented at the beginning of Section 3 or the more advanced algorithm of Section 3.3 with each rule switched off individually.

Early detection of non-identifiable instances

Worst-case performance of the search can be improved by detecting non-identifiable quantities directly based on the set of inputs before launching the search. The following theorem provides a sufficient criterion for non-identifiability.

Theorem 1. *Let G be a semi-Markovian graph, let $Q = P(\mathbf{Y} \mid do(\mathbf{X}), \mathbf{W})$ and let*

$$\mathbf{P} = \{P(\mathbf{A}_1 \mid do(\mathbf{B}_1), \mathbf{C}_1), \ldots, P(\mathbf{A}_n \mid do(\mathbf{B}_n), \mathbf{C}_n)\}.$$

Then Q is not identifiable from $\mathbf{P}$ in G via rules of Table 2 if

$$\mathbf{Y} \not\subseteq \bigcup_{i=1}^{n} \mathbf{A}_i,$$

Proof. Since $\mathbf{Y} \not\subseteq \bigcup_{i=1}^{n} \mathbf{A}_i$, there exists a variable $Y_j \in \mathbf{Y}$ such that none of the sets $\mathbf{A}_i$ contain it. No rule of Table 2 outputs a distribution $P(\mathbf{Y}' \mid \mathrm{do}(\mathbf{X}'), \mathbf{W}')$ such that some member of $\mathbf{Y}'$ would not already exist on the left-hand side of the input or additional input of the rule. Thus there is no sequence of rules that when applied to the available inputs $\mathbf{P}$ would result in a distribution of the form $P(Y_j, \cdot \mid \mathrm{do}(\cdot), \cdot)$. Thus there is no such sequence for $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$. □

In other words, Theorem 1 can be used to verify that the entire set $\mathbf{Y}$ of a target distribution $P(\mathbf{Y} \mid \cdot)$ cannot be constructed from the inputs. If this is the case, the target quantity is not identifiable.

Heuristics

During the search, we always expand one term at a time through the rules and store the newly identified distributions. In order for the search to perform fast, we need to decide which branches are the most promising and should therefore be expanded first. We can do this by defining a proximity function relating the source terms and the target query, and by always expanding the closest term first.

Our suggestion here is motivated by the way an educated person might apply do-calculus in a manual derivation. Our chosen proximity function h links the target distribution $P^t = P(\mathbf{A}_t \mid \mathrm{do}(\mathbf{B}_t), \mathbf{C}_t)$ and a source distribution $P^s = P(\mathbf{A}_s \mid \mathrm{do}(\mathbf{B}_s), \mathbf{C}_s)$ in the following way:

$$\begin{aligned} h(P^t, P^s) = {} & 10|\mathbf{A}_t \cap \mathbf{A}_s| + 5|\mathbf{B}_t \cap \mathbf{B}_s| + 3|\mathbf{C}_t \cap \mathbf{C}_s| - 2|\mathbf{A}_t \setminus \mathbf{A}_s| - 2|\mathbf{B}_t \setminus \mathbf{B}_s| \\ & - 2|\mathbf{B}_s \setminus \mathbf{B}_t| - |\mathbf{C}_t \setminus \mathbf{C}_s| - |\mathbf{C}_s \setminus \mathbf{C}_t|. \end{aligned}$$

Each input distribution and terms derived using the search are assigned into a priority queue, where the priority is determined by the value given by h. Distributions closer to the target are prioritized over other terms.

The weight 10 for the term $|\mathbf{A}_t \cap \mathbf{A}_s|$ indicates that having the correct response variables is considered as the first priority. Having the correct intervention is considered as the second priority (weight 5) and having the correct condition as the third priority (weight 3). The remaining terms in h penalize variables that are in the target distribution but not in the source distribution or vice versa. Again, variables that are intervened on are considered to be more important than conditioning variables.

3.3. The search algorithm

We take Algorithm 1 as our starting point and compile the results of Section 3.2 into a new search algorithm called do-search. This algorithm is capable of solving generalized identifiability problems (Row 8 in Table 1) while streamlining the search process through a heuristic search order and elimination of redundant rules and subsets. The pseudo-code for do-search is shown in Algorithm 2.

The algorithm begins by checking whether the query can be solved trivially without performing the search. This can happen if the target Q is a member of the set of inputs or if Theorem 1 applies. Next, we note that each input distribution in the set $\mathbf{P}$ is marked as unexpanded at the beginning of the search. Distributions in $\mathbf{P}$ are expanded one at a time by applying every rule of Table 2 in every possible way.

Algorithm 2 do-search

Input: Target $Q = P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$, a semi-Markovian graph G and a set of known distributions $\mathbf{P} = \{P_1, \ldots, P_n\}$.

Output: A formula F for Q in terms of $\mathbf{P}$ or NA

1: **if** $Q \in \mathbf{P}$, **return** Q
2: **if** target is non-identifiable by Theorem 1, **then return** NA
3: **let U** be the set of unexpanded distributions, initially $\mathbf{U} := \mathbf{P}$
4: **while** $\mathbf{U} \neq \emptyset$, **do**
5:   **let** P' be the unexpanded distribution closest to the target: $P' = \operatorname{argmax}_{P_i \in \mathbf{U}} h(Q, P_i)$
6:   **let M** be the set of rules of Table 2, without Rules 1±, such that the termination conditions of Table 3 do not hold with respect to P'.
7:   **let** $\mathbf{P}^*$ be the set of all distributions derived from P' using the rules in $\mathbf{M}$
8:   **for** each new candidate distribution $P^* \in \mathbf{P}^*$, **do**
9:     **if** P^* is already in $\mathbf{P}$, **then continue**
10:     **if** the validity conditions of Table 3 are not satisfied by P^*, **then continue**
11:     **if** an additional input is required that is not in $\mathbf{P}$, **then continue**
12:     **if** Rule 2± or 3± of Table 2 is applied and the corresponding d-separation Criterion 1 is not satisfied by G, **then continue**
13:     **if** $P^* = Q$, **then**
14:       Derive a formula F for Q by backtracking.
15:       **return** F
16:     Add P^* to $\mathbf{P}$, add P^* to $\mathbf{U}$
17:   Mark P' as expanded: remove P' from $\mathbf{U}$
18: **return** NA

The iteration over the unexpanded distributions $\mathbf{U}$ proceeds as follows (Lines 4–5). Each input distribution and terms derived from it are assigned into a priority queue, where the priority is determined by the value given by the proximity function h. Distributions closest to the target are expanded first. In the implementation, only the actual memory addresses of the distribution objects are placed into the queue. The set $\mathbf{P}$ is implemented as a hash table that serves as a container for all input distributions and those derived from them. Each new distribution is assigned a unique index that also serves as the hash function for this table. The distribution objects contained in the table are represented uniquely by three integers corresponding to the sets $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ of the general form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C})$. A distribution object also contains additional auxiliary information such as which rule was used to derive it, whether it is expanded or not and from which distribution it was obtained. This information is used to construct the derivation if the target is found to be identifiable.

Multiple distributions can share the same value of the proximity function h. In the case that multiple candidates share the maximal value, the one that was derived the earliest takes precedence. When the unexpanded distribution currently closest to the target is determined, the rules of Table 2 are applied sequentially for all valid subsets dictated by Table 3. When rules two and three of do-calculus are considered, the necessary d-separation criteria is checked from G (Line 12). For the chain rule, the presence of the required second input is also verified. The reverse lookup is implemented by using another hash table, where the hash function is based on the unique representation of each distribution object. The values contained in the table are the indices of the derived distributions. The same hash table is also used to ensure that we do not attempt to derive distributions again that have been previously found to be identifiable from the inputs.

We construct a set $\mathbf{M}$ of applicable rules for each unexpanded distribution P' using the termination conditions of Table 3 (Line 6). If all the necessary conditions have been found to hold for an applicable rule and a subset, the newly derived distribution P^* is added to the set of known distributions and placed into the priority queue as an unexpanded distribution. When the applicable rules and subsets have been exhausted for the current distribution P', the term is marked as expanded and removed from the queue (Line 17). If the target distribution is found at any point (Line 13), a formula is returned for it in terms of the original inputs. Alternatively, we can also continue deriving distributions to obtain different search paths to the target that can possibly produce different formulas for it. If instead we exhaust the set of unexpanded distributions by emptying the queue, the target is deemed non-identifiable by the search (Line 18).

We keep track of the rules that were used to derive each new distribution in the search. This allows us to construct a directed graph of the derivation where each root node is a member of the original input set $\mathbf{P}$ and their descendants are the distributions derived from them during the search. Each edge represents a manipulation of the parent node(s) to obtain the child node. For an identifiable target quantity, the formula F is obtained by backtracking the chain of manipulations recursively until the roots are reached (Line 14). The derivation of the example in the beginning of Section 3 depicted in Figure 2(c) can be efficiently found by applying this procedure.

We assess the worst case complexity of do-search in terms of the input graph size, which is the primary determining factor of the search time. Checks for separation and termination and validation conditions all run in polynomial time with respect to the number of vertices, but the number of distributions grows very rapidly. In a hypothetical absolute worst case scenario, the target is not identifiable, but every other distribution is. Supposing a graph with d vertices, we can determine how many distributions would have to be derived by the search in order to exhaust the search space. For a distribution of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C})$, any variable in $\mathbf{V}$ can either belong to the sets $\mathbf{A}, \mathbf{B}$ or $\mathbf{C}$ or not be present in the distribution, meaning that there are 4^d ways to categorize every variable. However, we must account for the fact that there must always be at least one variable in the set $\mathbf{A}$. There are 3^d ways to assign all the variables in such a way that no variable is a member of set $\mathbf{A}$. Thus the total number of distributions considered by do-search grows as $O(4^d)$. See the simulations of Section 5.1 for running time performance in more realistic settings.

3.4. Soundness and completeness properties

We are ready to establish some key theoretical properties of do-search. The first theorem considers the correctness of the search.

Theorem 2 (Soundness)**.** do-search *always terminates: if it returns an expression for the target Q, it is correct, if it returns NA then Q is not identifiable with respect to the rules of do-calculus and standard probability manipulations (in Table 2).*

Proof. Each new distribution is derived by using only well-defined manipulations as outlined by Table 3 and by ensuring that the required separation criteria hold in G when rules of do-calculus are concerned. It follows that if the search terminates and returns a formula for the target distribution, it was reached from the set input distributions through a sequence of valid manipulations. If do-search terminates as a result of Theorem 1, we are done. Suppose now that Theorem 1 does not apply. By definition, do-search enumerates every rule of Table 2 for every well-defined subset of Table 3. By Lemma 1, no distributions are left out by applying the termination criteria of Table 3. We know that if Rules 1$\pm$ of Table 3 are omitted, the distributions generated by these rules can be obtained by a combination of Rules 2$\pm$ and 3$\pm$. Furthermore, the order in which the distributions are expanded does not matter, as every possible manipulation is carried out nonetheless. The search will eventually terminate, since distributions that have already been derived are not added again to the set of unexpanded distributions and there are only finitely many ways to apply the rules of Table 2. $\square$

The following theorem provides a completeness result in connection to existing identifiability results. Since do-calculus has been shown to be complete with respect to (conditional) causal effect identifiability, z-identifiability, g-identifiability and transportability, it follows that do-search is complete for these problems as well.

Theorem 3 (Completeness)**.** *If* do-search *returns NA in the settings in Rows 1–4 and 6 in Table 1, then the query is non-identifiable.*

Proof. Do-calculus has been shown to be complete in these settings. The rules of probability calculus encode what is used in the algorithms as can be seen for example from the proofs of Theorem 7 and Lemmas 4–8 of Shpitser and Pearl (2006b). $\square$

It is not known whether the rules implemented in do-search are sufficient for other more general identifiability problems since it is conceivable that some additional rules might exist that would be required to achieve completeness. One such generalization is the inclusion of missing data in the causal model, which we present in Section 4. However, if one were to show that do-calculus (or any other set of rules included in do-search) is complete for some special case of the generalized identifiability problem, then do-search would be complete for this problem as well. In the following sections we will use the term "identifiable by do-search" to refer to causal queries that can be identified by do-search.

4. Extension to missing data problems

The SCM framework can be extended to describe missing data mechanisms. For each variable V_i, two special vertices are added to the causal graph. The vertex V_i^* is the observed proxy variable which is linked to the true variable V_i via the missingness mechanism (Little and Rubin 1986; Mohan *et al.* 2013):

$$V_i^* = \begin{cases} V_i, & \text{if } R_{V_i} = 1, \\ \text{NA}, & \text{if } R_{V_i} = 0, \end{cases} \tag{2}$$

where NA denotes a missing value and R_{V_i} is called the response indicator (of V_i). In other words, the variable V_i^* that is actually observed matches the true value V_i if it is not missing ($R_{V_i} = 1$). We note that in this formulation, each true variable has its own response indicator, meaning that we do not consider shared indicators between variables or multiple indicators for a single variable. Figure 11 in Section 6.4 depicts some examples of graphs containing missing data mechanisms. Furthermore, if there is no missingness associated with a given variable V_i meaning that it is fully observed, the corresponding response indicator R_{V_i} always has the value 1. The omission of a proxy variable and a response indicators of a specific variable from a graph encodes the assumption that the variable in question if fully observed. Note that intervention nodes are added for true variables and response indicators but not for proxy variables. On a symbolic level one could intervene on proxy variables, however we are only interested in interventions that keep Equation 2 intact.

The observed vertices of the causal diagram can be partitioned into three categories

$$\mathbf{V} = \mathbf{V}^t \cup \mathbf{V}^* \cup \mathbf{V}^r,$$

where $\mathbf{V}^t$ is the set of true variables, $\mathbf{V}^*$ is the set of proxy variables and $\mathbf{V}^r$ is the set of response indicators. For any subset $\mathbf{Z} \subset \mathbf{V}$ we define the same partition via $\mathbf{Z}^t = \mathbf{Z} \cap \mathbf{V}^t$, $\mathbf{Z}^* = \mathbf{Z} \cap \mathbf{V}^*$ and $\mathbf{Z}^r = \mathbf{Z} \cap \mathbf{V}^r$.

The definition of the response indicator connects a proxy variable and a true variable. Typically this connection is only of interest for those variables V_i that have missing data, meaning that $P(R_{V_i} = 1) < 1$. Furthermore, we often utilize a proxy variable corresponding to a specific true variable and conversely, a true variable corresponding to a specific proxy variable. We define this correspondence explicitly in the following way

$$\mathbf{Z}^{(t \to *)} = \{V_i^* \in \mathbf{V}^* \mid V_i \in \mathbf{Z}^t, P(R_{V_i} = 1) < 1\},$$
$$\mathbf{Z}^{(* \to t)} = \{V_i \in \mathbf{V}^t \mid V_i^* \in \mathbf{Z}^*, P(R_{V_i} = 1) < 1\}.$$

Similarly, given a set $\mathbf{Z}$ we often require the set of the response indicators that define the missingness mechanism for the true variables that are member of $\mathbf{Z}$. This set is defined as follows

$$\mathbf{R}_{\mathbf{Z}} = \{R_{V_i} \in \mathbf{V}^r \mid V_i \in \mathbf{Z}^t, P(R_{V_i} = 1) < 1\}.$$

It is important to note the difference between the sets $\mathbf{Z}^r$ and $\mathbf{R}_{\mathbf{Z}}$; the first set denotes the set of response indicators that are members of $\mathbf{Z}$ while the second gives the corresponding response indicators for the true variables that are member of $\mathbf{Z}$.

Our method is also capable of processing queries when the causal graph contains missing data mechanisms where the sets $\mathbf{A}_i, \mathbf{B}_i$ and $\mathbf{C}_i$ of some of the input distributions may be restricted to contain observed variables in $\mathbf{V}^* \cup \mathbf{V}^r$. An active response indicator $R_{V_i} = 1$ is denoted by $R^1_{V_i}$. Similarly, for sets of response indicators $\mathbf{R}^1_{\mathbf{Z}}$ denotes that all indicators in the set are active. Proxy variables are not explicitly shown in graphs for clarity.

Determining identifiability is challenging under missing data. As evidence of this, even some non-interventional queries require the application of do-calculus (Mohan and Pearl 2018). Furthermore, the rules used in the search of Table 2 are no longer sufficient and deriving the desired quantity necessitates the use of additional rules that stem from the definition of the proxy variables and the response indicator. Each new true variable also has a higher impact on computational complexity, since the corresponding response indicator and proxy variable are always added to the graph as well. Table 4 extends the set of rules of Table 2 to missing data problems by providing manipulations related to the missingness mechanism. Rules 7± and 8± perform conditioning using the chain rule. These rules are necessary in the case that set $\mathbf{Y}$ contains missing data mechanisms that have been enabled and thus cannot be marginalized over by using Rule 5.

Rule	Additional Input	Output	Description
1+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W})$	Insertion of observations
1−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	Deletion of observations
2+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W} \setminus \mathbf{Z})$	Obs. to action exchange
2−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X} \setminus \mathbf{Z}), \mathbf{Z}, \mathbf{W})$	Action to obs. exchange
3+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}, \mathbf{Z}), \mathbf{W})$	Insertion of actions
3−		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X} \setminus \mathbf{Z}), \mathbf{W})$	Deletion of actions
4		$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Marginalization
5		$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W})$	Conditioning
6+	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	Chain rule multiplication
6−	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Y}, \mathbf{W})$	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Chain rule multiplication
7+	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{Z}, \mathbf{W})$	Chain rule conditioning (numerator)
7−	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}, \mathbf{Y} \setminus \mathbf{Z})$	$P(\mathbf{Y} \setminus \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Chain rule conditioning (numerator)
8+	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W}, \mathbf{Y})$	Chain rule conditioning (denominator)
8−	$P(\mathbf{Y}, \mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z})$	$P(\mathbf{Z} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Chain rule conditioning (denominator)
9+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{R}_{\mathbf{Z}}, \mathbf{R}^1_{\mathbf{Z}})$	Enable response indicators
9−		$P(\mathbf{Y} \setminus \mathbf{R}_{\mathbf{Z}}, \mathbf{R}^1_{\mathbf{Z}} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Enable response indicators
10+		$P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W} \setminus \mathbf{Z}^*, \mathbf{Z}^{(* \to t)})$	Proxy variable exchange
10−		$P(\mathbf{Y} \setminus \mathbf{Z}^*, \mathbf{Z}^{(* \to t)} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$	Proxy variable exchange

Table 4: Extended set of rules for missing data problems used to manipulate input distributions of the form $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$. Rules $1\pm, 2\pm, 3\pm, 4, 5$ and $6\pm$ are the same as in Table 2. For Rules $7\pm$ and $8\pm$, the additional input has also been identified. The sets $\mathbf{Y}, \mathbf{X}$ and $\mathbf{W}$ are disjoint. Sets $\mathbf{Y}$ and $\mathbf{W}$ may contain true variables, proxy variables and response indicators. Set $\mathbf{X}$ may only contain true variables and response indicators. The roles of the sets $\mathbf{Z}$ and $\mathbf{R}_{\mathbf{Z}}$ depend on the rule being applied (see Table 5).

Rules $9\pm$ are used to enable response indicators, which then facilitates the use of Rules $10\pm$. These last two rules exchange proxy variables to their true counterparts when the corresponding response indicators are enabled. For example, under the conditions specified in Table 5, Rule 9+ can be applied on $P(Y, X^* \mid R_X)$ to first obtain $P(Y, X^* \mid R^1_X)$ by enabling R_X. Then, Rule 10+ can applied to this distribution to obtain $P(Y, X \mid R^1_X)$ by exchanging X^* for X.

Similarly to Table 3, Table 5 outlines the valid subsets $\mathbf{Z}$ for applying the extended rules of Table 4. A major difference to the original validity and termination conditions is the addition of the missing data condition that outlines the additional requirements that must be satisfied when missingness mechanisms are present. For the rules that are shared by Tables 2 and 4, the missing data condition ensures that a true variable and its proxy counterpart never appear in the same term at the same time. For example, we cannot add an intervention on X to $P(X^*)$. It also ensures that we do not carry out summation over enabled response indicators in the case of rules 4 and 5. When applying Rules $9\pm$, the condition also ensures that we do not attempt to enable a response indicator that is already enabled. For Rules $10\pm$, the conditions guarantee that a proxy can only be exchanged to its true counterpart if its corresponding response indicator is enabled and present in the input term.

Additional termination conditions also apply to the new rules and their correctness is easily verified.

Lemma 2. *Let G be a semi-Markovian graph and let $\mathbf{Y}, \mathbf{X}$ and $\mathbf{W}$ be disjoint subsets of $\mathbf{V}$. Then all of the following are true:*

Rule	Validity cond.	Missing data condition	Term. cond.
1+	$\mathbf{Z} \cap \mathbf{T} = \emptyset$	$\mathbf{Z} \cap \left(\mathbf{T}^{(*\to t)} \cup \mathbf{T}^{(t\to *)} \cup \mathbf{Z}^{(*\to t)} \cup \mathbf{Z}^{(t\to *)}\right) = \emptyset$	
1−	$\mathbf{Z} \subseteq \mathbf{W}$		$\mathbf{W} = \emptyset$
2+	$\mathbf{Z} \subseteq \mathbf{W}$	$\mathbf{Z} \cap \mathbf{W}^* = \emptyset$	$\mathbf{W} = \emptyset$
2−	$\mathbf{Z} \subseteq \mathbf{X}$		$\mathbf{X} = \emptyset$
3+	$\mathbf{Z} \cap \mathbf{T} = \emptyset$	$\mathbf{Z} \cap \left(\mathbf{T}^{(*\to t)} \cup \mathbf{T}^{(t\to *)} \cup \mathbf{Z}^{(*\to t)} \cup \mathbf{Z}^{(t\to *)}\right) = \emptyset$	
3−	$\mathbf{Z} \subseteq \mathbf{X}$		$\mathbf{X} = \emptyset$
4	$\mathbf{Z} \subset \mathbf{Y}$	$\mathbf{Z} \cap (\mathbf{R}^a \cap \mathbf{Y}) = \emptyset$	$\lvert\mathbf{Y}\rvert = 1$
5	$\mathbf{Z} \subset \mathbf{Y}$	$(\mathbf{Y} \setminus \mathbf{Z}) \cap (\mathbf{R}^a \cap \mathbf{Y}) = \emptyset$	$\lvert\mathbf{Y}\rvert = 1$
6+	$\mathbf{Z} \subseteq \mathbf{W}$		$\mathbf{W} = \emptyset$
6−	$\mathbf{Z} \cap \mathbf{T} = \emptyset$	$\mathbf{Z} \cap \left(\mathbf{T}^{(*\to t)} \cup \mathbf{T}^{(t\to *)} \cup \mathbf{Z}^{(*\to t)} \cup \mathbf{Z}^{(t\to *)}\right) = \emptyset$	
7+		$\mathbf{Z} \subset \mathbf{Y}$	$\lvert\mathbf{Y}\rvert = 1$
7−		$\mathbf{Z} \subset \mathbf{Y}$	$\lvert\mathbf{Y}\rvert = 1$
8+		$\mathbf{Z} \cap \mathbf{T} = \emptyset$	
8−		$\mathbf{Z} \subseteq \mathbf{W}$	$\mathbf{W} = \emptyset$
9+		$\mathbf{R}_\mathbf{Z} \subseteq \mathbf{W}^r, \mathbf{R}_\mathbf{Z} \cap \mathbf{R}^a = \emptyset$	$\mathbf{W}^r = \emptyset$
9−		$\mathbf{R}_\mathbf{Z} \subseteq \mathbf{Y}^r,\ \mathbf{R}_\mathbf{Z} \cap \mathbf{R}^a = \emptyset$	$\mathbf{Y}^r = \emptyset$
10+		$\mathbf{Z}^* \subseteq \mathbf{W}^*, \mathbf{R}_{\mathbf{Z}^{(*\to t)}} \subseteq \mathbf{R}^a, \mathbf{R}_{\mathbf{Z}^{(*\to t)}} \subseteq \mathbf{W}^r$	$\mathbf{R}^a = \emptyset$
10−		$\mathbf{Z}^* \subseteq \mathbf{Y}^*,\ \mathbf{R}_{\mathbf{Z}^{(*\to t)}} \subseteq \mathbf{R}^a, (\mathbf{R}_{\mathbf{Z}^{(*\to t)}} \subseteq \mathbf{W}^r \text{ or } \mathbf{R}_{\mathbf{Z}^{(*\to t)}} \subseteq \mathbf{Y}^r)$	$\mathbf{R}^a = \emptyset$

Table 5: The conditions for the enumerated subset $\mathbf{Z}$ for applying the rules of Table 4 to a term in the input column. Here $\mathbf{T} = \mathbf{Y} \cup \mathbf{X} \cup \mathbf{W}$ and the sets $\mathbf{Y}, \mathbf{X}$ and $\mathbf{W}$ are those present in the input term $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$. Active response indicators of the input are denoted by $\mathbf{R}^a$. For Rules $6\pm, 7\pm$ and $8\pm$, the conditions specify valid variables of the second required term. Validity conditions for Rules $1\pm, 2\pm, 3\pm, 4, 5$ and $6\pm$ are the same as in Table 3.

(i) *If* $\lvert\mathbf{Y}\rvert = 1$*, then Rules* $7\pm$ *of Table 4 cannot be used.*

(ii) *If* $\mathbf{W} = \emptyset$*, then Rule* $8-$ *of Table 4 cannot be used.*

(iii) *If* $\mathbf{W}^r = \emptyset$ *then Rule* $9+$ *of Table 4 cannot be used.*

(iv) *If* $\mathbf{Y}^r = \emptyset$ *then Rule* $9-$ *of Table 4 cannot be used.*

(v) *If* $\mathbf{R}^a = \emptyset$*, then Rules* $10\pm$ *of Table 4 cannot be used.*

Proof. For (i), the set $\mathbf{Y}$ only has a single vertex, so it cannot have a non-empty subset. For (ii), the set $\mathbf{W}$ is empty so no subset $\mathbf{Z} \subset \mathbf{W}$ can exist for the second input. For (iii), and the set $\mathbf{W}^r$ is empty so no assignment to value 1 can be performed. Similarly for (iv), the set $\mathbf{Y}^r$ is empty so no assignment to value 1 can be performed. For (v) the set of active response indicators $\mathbf{R}^a$ is empty, so no transformation from proxy variables to true variables via the missingness mechanism in Equation 2 can take place. □

The task of selecting a suitable heuristic becomes more difficult when missing data are involved with the identifiability problem. The heuristic approach of Section 3.2 is no longer directly applicable due to the relation between proxy variables, response indicators and true variables. The proximity function considers X and X^* as entirely different variables despite their connection and does not prefer the inclusion of response indicators. If the heuristic is applied as such, the search path will often involve a large number of manipulations which in turn leads to complicated expressions. For these reasons we do not apply a heuristic to missing data problems, but expand terms in the order in which they were identified. The improvements described in Section 3.2 still apply.

It is straightforward to adapt do-search to the new extended set of rules. In the pseudocode shown in Algorithm 2, we simply replace all references to Tables 2 and 3 by references to Tables 4 and Tables 5, respectively. When the validity condition is checked, we also verify that the missing data condition holds. Lemma 2 guarantees the correctness of the new termination criteria. Theorem 1 is also valid when the sets $\mathbf{A}_i$ are replaced by $\mathbf{A}_i \cup \mathbf{A}_i^{(*\rightarrow t)}$, since it may be possible to exchange some proxy variable to a true variable that is present in the set $\mathbf{Y}$ of the target $P(\mathbf{Y} \mid \mathrm{do}(\mathbf{X}), \mathbf{W})$.

5. The dosearch package

We implemented do-search (Algorithm 2) in C++ and constructed an R interface using the **Rcpp** package (Eddelbuettel and François 2011). This interface is provided by the R package **dosearch**. Calling the search from R is straightforward via the primary function that carries the name of package.

```
dosearch(data, query, graph,
  transportability, selection_bias, missing_data,
  control)
```

The required inputs of the function are `data`, `query` and `graph`. Parameter `data` is used to encode the set $\mathbf{P}$ of known input distributions of Algorithm 2 as a character string, where each distribution is separated by a new line. For example, if we have access to a set of distributions $\mathbf{P} = \{P(W), P(Y \mid X), P(Z \mid \mathrm{do}(X), W)\}$, we would write

```
R> data <- "
+     P(W)
+     P(Y|X)
+     P(Z|do(X),W)
+  "
```

The do($\cdot$)-operator can either precede or succeed conditioning variables, but it must appear only once in a given term, meaning that expressions such as `P(Y|do(A),B,do(C))` are not allowed, but should instead be given as `P(Y|B,do(A,C))` or `P(Y|do(A,C),B)`. If variable sets are desired, each member of the set has to be included explicitly.

Parameter `query` is used to describe the target Q of Algorithm 2 as a character string, similarly as the `data`. If we are interested in identifying $P(Y \mid \mathrm{do}(X), W)$ we would write

```
R> query <- "P(Y|do(X),W)"
```

Instead of describing distributions via text, it is also possible to use the following structure that encodes the role of each variable via a numeric vector:

```
R> query <- c(Y = 0, X = 1, W = 2)
```

Given a distribution of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C})$ and a variable V, a value 0 means that $V \in \mathbf{A}$, value 1 means that $V \in \mathbf{B}$ and value 2 means that $V \in \mathbf{C}$. This format can also be used to input `data` as a list of numeric vectors:

```
R> data <- list(
+    c(W = 0),
+    c(Y = 0, X = 2),
+    c(Z = 0, X = 1, W = 2)
+  )
```

Finally, `graph` encodes the semi-Markovian graph G of the causal model as a character string with each edge on its own line. A directed edge from X to Y is given as `X -> Y` and a bidirected edge between X and Y is given as `X <-> Y`. Intervention nodes should not be given explicitly, since they are added automatically after calling `dosearch`. Furthermore, only vertices with incoming or outgoing edges should be included in `graph`. A variable with no connected edges can still appear in the input distributions and is automatically added to the graph as well. As an example, we can encode the graph of Figure 2(a) with an added bidirected edge between X and Y as follows

```
R> graph <- "
+    X -> Y
+    Z -> X
+    Z -> Y
+    X <-> Y
+  "
```

Alternatively, one may use **igraph** graphs (Csardi and Nepusz 2006) in the syntax of the **causaleffect** package or DAGs created using the **dagitty** package.

```
R> library("igraph")
R> graph <- graph.formula(X -+ Y, Z -+ X, Z -+ Y, X -+ Y, Y -+ X)
R> graph <- set.edge.attribute(graph, "description", 4:5, "U")
R> library("dagitty")
R> graph <- dagitty("dag{X -> Y; Z -> X; Z -> Y; X <-> Y}")
```

The next two optional parameters of `dosearch`, `transportability` and `selection_bias`, are used to denote those vertices of G that should be understood as either transportability nodes or selection bias nodes, respectively. Providing these parameters may increase search performance in relevant problems. Both of these parameters should be given as character strings, where individual variables are separated by a comma, for example `transportability = "S,T"`. Parameter `missing_data`, as the name suggests, is used to define missingness mechanisms 2 as a character string, where individual mechanisms are separated by a comma. In order to describe that R_X is the response indicator of X we would write `R_X : X`, which also implicitly defines that `X*` is the proxy variable of `X`. Proxy variables do not need to be manually described in `graph` as they are automatically constructed based on the `missing_data` argument.

The list `control` can be used to set various additional parameters that are not directly related to the identifiability problem itself, but more so to the output of the search and other auxiliary details, such as benchmarking and obtaining derivations such as Figure 2(c). One such control parameter determines whether to use the search heuristic or not (`heuristic = FALSE` by default). Documentation of the **dosearch** package contains detailed information on the full list of control parameters.

The return object of `dosearch` is a list with three components by default. The first component, `identifiability`, is a logical value that takes the value `TRUE` when the target distribution described by `query` is identifiable from the inputs of `data`. The second component, `formula`, is a character string describing the target distribution in terms of the inputs in LaTeX syntax if the target is identifiable. Otherwise this component is just an empty character string. The third component `call` contains the arguments of the original function call.

5.1. Simulations

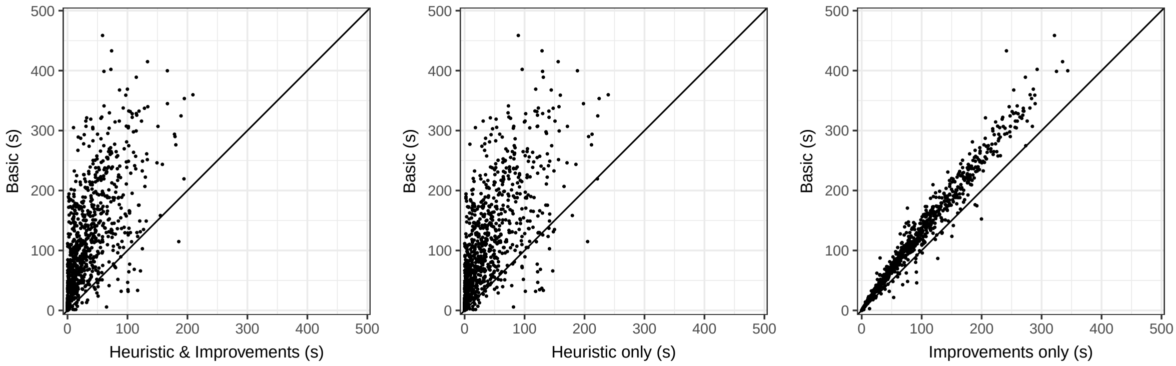
Here we report the results of a simulation study to assess the running time performance of dosearch and the impact of the search space reduction techniques as well as the search heuristic outlined in Section 3.2.

Our synthetic simulation scenario consisted of 1000 semi-Markovian causal graphs of 10 vertices that were generated at random by first generating a random topological order of the vertices followed by a random lower triangular adjacency matrices for both directed and bidirected edges. Graphs without a directed path from X to Y were discarded. We sampled sequentially input distributions of the form $P(\mathbf{A} \mid \mathrm{do}(\mathbf{B}), \mathbf{C})$ at random by generating disjoint subsets such that $\mathbf{A}$ is always non-empty. This was continued until the target quantity $P(Y \mid \mathrm{do}(X))$ was found to be identifiable by the search. Then for each graph, we recorded the search times for the specific set of inputs that first resulted in the query to be identified and for the last set such that the target was non-identifiable. In other words, each graph generates two simulation instances, one for an identifiable query and one for a non-identifiable query. This setting directly corresponds to the setting of partially overlapping experimental data sets discussed in Section 2.2 for which no other algorithmic solutions exist.

To understand the impact of the search heuristic and the various improvements, we compare four different search configurations: the basic do-search without the search heuristic or improvements[1], one that only uses the search heuristic, one that only uses the improvements of Section 3.2 and one that uses them both.

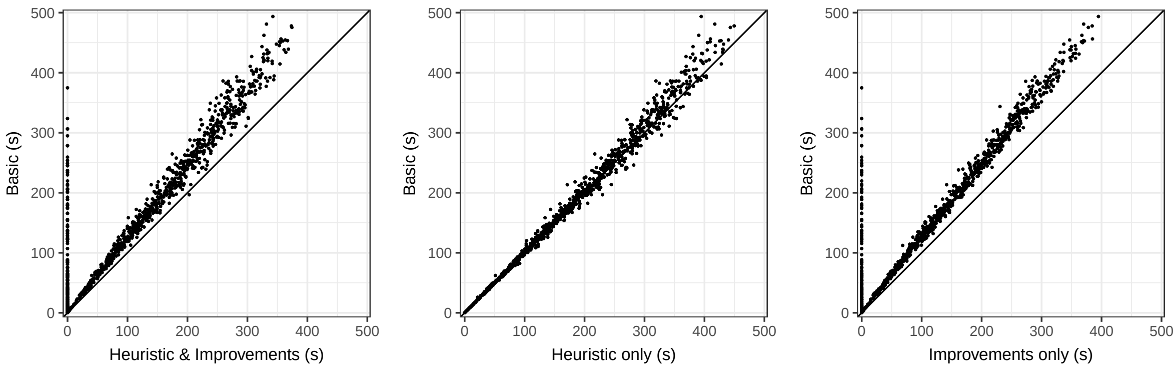
Figure 4 shows the search times of the configurations compared to the basic configuration for identifiable instances. Most importantly, a vast majority of instances (96%) are solved faster than the basic configuration when both heuristics and improvements are used. The average search time with both heuristics and improvements enabled was 31.5 seconds and 80.2 seconds for the basic configuration. The search heuristic provides the greatest benefit for these instances as can be seen from Figure 4(b). Using a heuristic can sometimes hinder performance by leading the search astray and by causing additional computational steps through the evaluation of the proximity function. For example, there is a small number of instances where the search is over ten times slower than the basic configuration when using a heuristic. Fortunately, there are several instances in the opposite direction, where the heuristic provides over one hundred fold reduction in search time. Curiously, even using the improvements sometimes results in slower search times. This is most likely due to the elimination of Rule 1 of do-calculus, since it may be the case that the basic search is able to use this rule to reach the target distribution faster. More importantly, Figure 4(c) shows that the improvements clearly benefit the search. Furthermore, the benefit tends to increase as the instances get harder.

[1]In this configuration, terms are expanded in the order they were identified; the conditions in Table 3 are not checked.



(a) (b) (c)

Figure 4: Scatter plots of the search times from identifiable instances under different search configurations compared to the basic do-search without a heuristic or improvements.



(a) (b) (c)

Figure 5: Scatter plots of the search times from non-identifiable instances under different search configurations compared to the baseline configuration.

Figure 5 shows the search times of the configurations for non-identifiable instances. Relying only on a search heuristic provides no benefit here, as expected. The improvements to the search are most valuable for these instances, and in this scenario every non-identifiable instance was solved faster than baseline using the improvements, and when applied with the heuristic only one non-identifiable instance was slower than baseline. The average search time with both heuristic and improvements enabled was 134.6 seconds and 182.5 seconds for the basic configuration. The almost zero second instances are a result of Theorem 1 when no search has to be performed in order to determine the instance to be non-identifiable. The benefit of the improvements tends to increase as the instances get harder also for these instances.

Finally we examined the average run time performance of do-search, with all improvements and heuristics enabled. We replicated the previously described simulation scenario with the same number of instances (1000) for graphs with up to 10 vertices. Figure 6 shows the boxplots of search times on a log-scale for graphs of different size, including both identifiable and non-identifiable instances. Note that for every graph size there are a number of easily

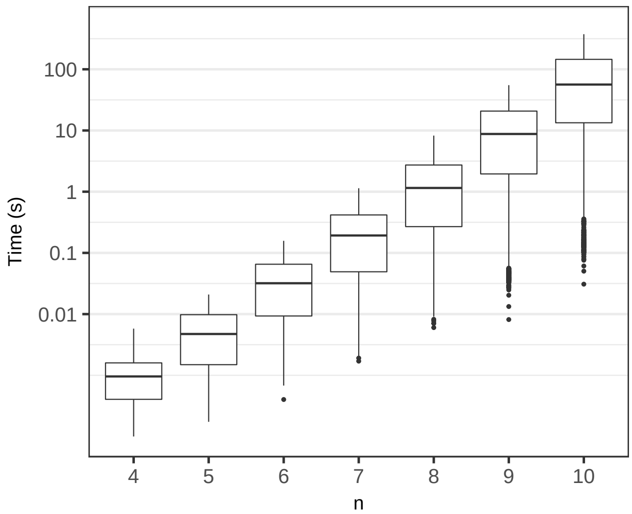


Figure 6: Boxplots of search times for both identifiable and non-identifiable instances in graphs of $n = 4, \ldots, 10$ vertices. The vertical axis uses a logarithmic scaling. Instances where the search time was less than 10^{-4} seconds were omitted for clarity.

solvable instances that show up as outliers in this plot. Instances with graphs of 10 vertices are solved routinely in under 100 seconds. In this plot, the running times increase exponentially with increasing graph size (or number of variables).

6. New causal effect identification results

We present a number of results for various identifiability problems to showcase the versatility of do-search with the accompanying R code for some specific examples.

6.1. Multiple data sources with partially overlapping variable sets

Earlier generalizations of the identifiability problem assume nested experiments or entire distributions with the exception of surrogate outcome identifiability (Tikka and Karvanen 2019) which also has its own intricate set of assumptions regarding the available distributions. None of these assumptions are needed in do-search and it can be used to solve identifiability problems from completely arbitrary collections of input distributions.

We showcase identifiability from multiple data sources by three examples. The first example is the human resource management problem presented in the introduction and shown in Figure 1. The question of interest was the identifiability of $P(Y \mid \mathrm{do}(X))$ from the data sources $P(Y, B, E, X)$ and $P(A, B, X)$. The answer can be provided with the following lines of R code:

```
R> library("dosearch")
R> data <- "
+    p(y,b,e,x)
+    p(a,b,x)
+  "
```

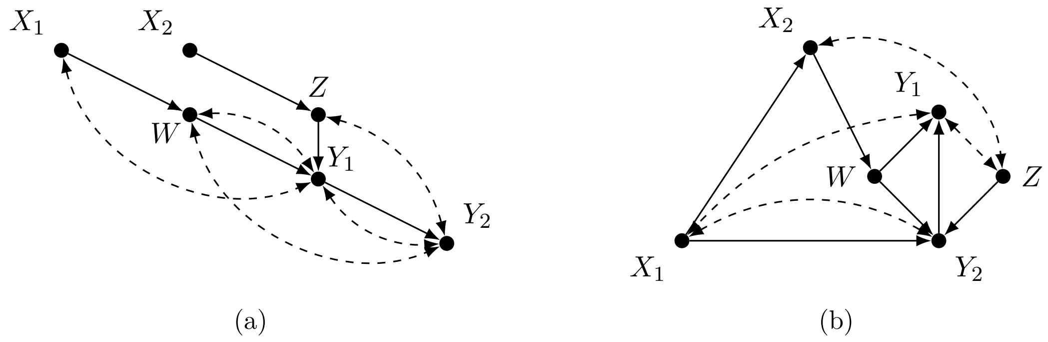


Figure 7: Graphs for the examples on identifiability problems combining both observational and experimental distributions.

```
R> query <- "p(y|do(x))"
R> graph <- "
+     e -> x
+     e -> y
+     a -> b
+     a -> x
+     x -> b
+     x -> y
+     b -> y
+   "
R> dosearch(data, query, graph, control = list(heuristic = TRUE))
```

```
\sum_{b,a}\left(p(a)\left(p(b|x,a)\sum_{e}
  \left(p(e)p(y|x,b,e)\right)\right)\right)
```

The result means that the causal effect is identifiable and the returned formula is

$$\sum_{B,A} P(A)P(B \mid X, A) \sum_{E} P(E)P(Y|X, B, E).$$

By running an additional line of code

```
R> dosearch(data, "p(y,b,e,x,a)", graph, control = list(heuristic = TRUE))
```

```
The query p(y,b,e,x,a) is non-identifiable.
```

we learn that the joint distribution $P(Y, B, E, X, A)$ is not identifiable. This rules out the possibility to solve the problem with **causaleffect** which requires the joint distribution as an input.

In the second example we consider identifiability of $P(Y_1, Y_2 \mid \mathrm{do}(X_1, X_2))$ in the graph of Figure 7(a) from $P(\mathbf{V})$, $P(Y_1, Y_2 \mid \mathrm{do}(X_1), Z, W, X_2)$, $P(W \mid \mathrm{do}(X_1, X_2))$ and $P(Z \mid \mathrm{do}(X_2))$. The target quantity is identifiable and do-search produces the following formula for it

$$\sum_{Z,W} P(Y_1, Y_2 \mid \mathrm{do}(X_1), Z, W, X_2)P(Z \mid \mathrm{do}(X_2))P(W \mid \mathrm{do}(X_2, X_1))$$

In the third example we consider identifiability of $P(Y_1, Y_2 \mid \mathrm{do}(X_1, X_2))$ in the graph of Figure 7(b) from $P(\mathbf{V})$, $P(Y_1 \mid \mathrm{do}(X_1), Y_2, W, Z, X_2)$, $P(X_2, W \mid \mathrm{do}(X_1))$, $P(X_2 \mid \mathrm{do}(X_1, W))$, $P(Y_2 \mid \mathrm{do}(X_1), Z, W, X_2)$, $P(Y_2 \mid \mathrm{do}(Z), X_1, W, X_2)$, and $P(Y_1, Y_2 \mid \mathrm{do}(Z), W, X_1, X_2)$. Again, the target quantity is identifiable and do-search outputs the following formula

$$\sum_{W} \Bigg(P(W \mid \mathrm{do}(X_1), X_2) \sum_{X_2} P(X_2 \mid \mathrm{do}(X_1, W)) \times \\ \frac{\sum_Z P(X_2, W, Z \mid X_1) P(Y_1, Y_2 \mid \mathrm{do}(X_1), X_2, W, Z)}{\sum_{Y_1', Y_2', Z} P(X_2, W, Z \mid X_1) P(Y_1', Y_2' \mid \mathrm{do}(X_1), X_2, W, Z)} \Bigg).$$

This example shows that a heuristic approach can also help us to find shorter formulas. If we run do-search again without the heuristic in this instance, the output formula is instead

$$\sum_{W,Z} \Bigg(P(Z) P(W \mid X_2, X_1, Z) \sum_{X_2} P(X_2 \mid X_1, Z) \sum_{Y_2} P(Y_2 \mid \mathrm{do}(X_1), X_2, W, Z) \times \\ P(Y_1 \mid \mathrm{do}(X_1), X_2, Y_2, W, Z) \frac{P(Y_2 \mid \mathrm{do}(X_1), X_2, W, Z) P(Y_1 \mid \mathrm{do}(X_1), X_2, Y_2, W, Z)}{\sum_{Y_2'} P(Y_2' \mid \mathrm{do}(X_1), X_2, W, Z) P(Y_1 \mid \mathrm{do}(X_1), X_2, Y_2', W, Z)} \Bigg).$$

We can run these examples in R by writing

```
R> data <- "
+    p(x_1,y_1,x_2,y_2,z,w)
+    p(y_1,y_2|z,w,x_2,do(x_1))
+    p(y_2|y_1,z,w,x_2,do(x_1))
+    p(w|do(x_1,x_2))
+    p(z|do(x_2))
+  "
R> query <- "p(y_1,y_2|do(x_1,x_2))"
R> graph <- "
+    z -> y_1
+    w -> y_1
+    y_1 -> y_2
+    x_2 -> z
+    x_1 -> w
+    y_1 <-> x_1
+    y_1 <-> y_2
+    y_2 <-> z
+    y_1 <-> w
+    y_2 <-> w
+  "
R> dosearch(data, query, graph, control = list(heuristic = TRUE))
```

```
\sum_{z,w}\left(p(y_1,y_2|do(x_1),z,w,x_2)
  \left(p(z|do(x_2))p(w|do(x_2,x_1))\right)\right)
```

and

```
R> data <- "
+    p(x_1,y_1,x_2,y_2,z,w)
+    p(y_1,y_2|w,x_1,x_2,do(z))
+    p(y_1|y_2,w,z,x_2,do(x_1))
+    p(y_2|x_1,w,x_2,do(z))
+    p(x_2,w|do(x_1))
+    p(x_2|do(x_1,w))
+    p(y_2|z,w,x_2,do(x_1))
+  "
R> query <- "p(y_1,y_2|do(x_1,x_2))"
R> graph <- "
+    y_2 -> y_1
+    w -> y_1
+    x_1 -> x_2
+    x_1 -> y_2
+    z -> y_2
+    w -> y_2
+    x_2 -> w
+    x_1 <-> y_1
+    x_1 <-> y_2
+    y_1 <-> z
+    x_2 <-> z
+  "
R> dosearch(data, query, graph, control = list(heuristic = TRUE))
```

```
\sum_{w}\left(p(w|do(x_1),x_2)\sum_{x_2}\left(p(x_2|do(w,x_1))
  \frac{\sum_{z}\left(p(x_2,w,z|x_1)p(y_1,y_2|do(x_1),x_2,w,z)\right)}
  {\sum_{y_1,y_2} \sum_{z}\left(p(x_2,w,z|x_1)p(y_1,y_2|do(x_1),x_2,w,z)
  \right)}\right)\right)
```

6.2. Combining transportability and selection bias

Input distributions that originate from multiple sources while being simultaneously affected by selection bias can be considered with do-search. This kind of problem cannot be solved with algorithms RC or $\mathrm{TR}^{\mathrm{mz}}$ of Table 1. As an example we consider one source domain and a target domain with two input data sets: a biased distribution $P(X, Y, Z \mid S)$ from the target domain and an unbiased experimental distribution $P(Y, Z \mid \mathrm{do}(X), T)$ from the source domain. We evaluate the query $P(Y \mid \mathrm{do}(X))$ in the graph of Figure 8 using these inputs. In the figure transportability node T is depicted as a gray square and selection bias node S is depicted as an open double circle. The query is identifiable and do-search outputs the following formula for it

$$P(Y \mid \mathrm{do}(X)) = \sum_{Z} P(Y \mid \mathrm{do}(X), Z, T) \sum_{Y'} P(Z, Y' \mid X, S).$$

In R we may write

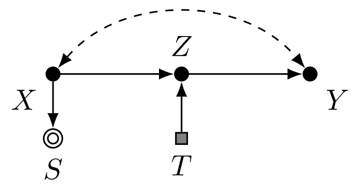


Figure 8: Graph that contains both selection bias and transportability nodes.

```
R> data <- "
+    p(x,z,y|s)
+    p(y,z|t,do(x))
+  "
R> query <- "p(y|do(x))"
R> graph <- "
+    x -> z
+    z -> y
+    x -> s
+    t -> z
+    x <-> y
+  "
R> dosearch(data, query, graph,
+    transportability = "t", selection_bias = "s"
+    control = list(heuristic = TRUE))
```

```
\sum_{z}\left(p(y|do(x),z,t)\sum_{y}p(z,y|x,s)\right)
```

6.3. Recovering from multiple sources of selection bias

We present an example where bias originates from two sources with two input data sets: a distribution affected by both biasing mechanisms $P(X, Y, Z, W_1, W_2 \mid S_1, S_2)$ and a distribution affected only by a single bias source $P(Z \mid S_1)$. We evaluate the query $P(Y \mid \mathrm{do}(X))$ in the graph of Figure 9 using the inputs. The query is identifiable and the following formula is

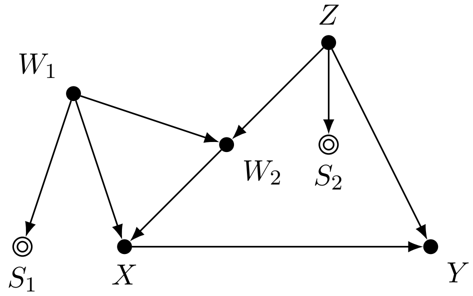


Figure 9: Graph where two selection bias nodes are present.

obtained by do-search

$$\sum_{Z} P(Z \mid S_1) P(Y \mid X, Z, W_1, W_2, S_1, S_2).$$

The result can be obtained in R as follows. In this case, slightly nicer expression is produced when not using the heuristic than when using the heuristic.

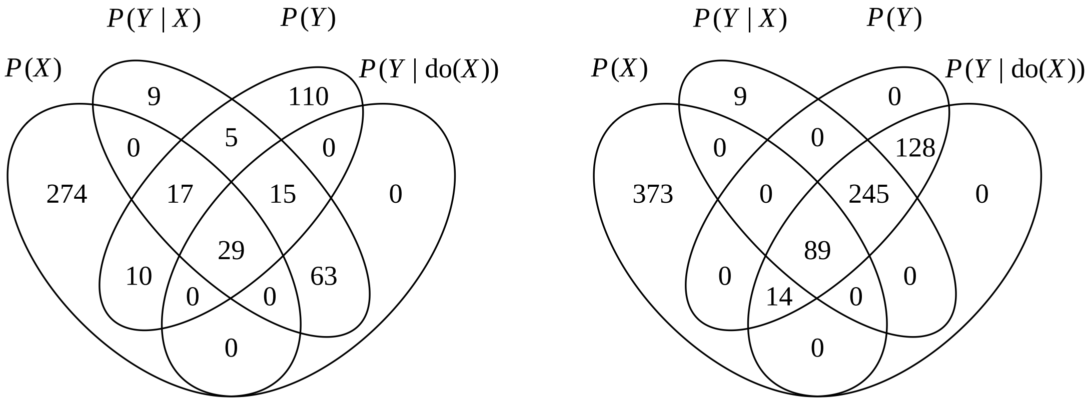
```
R> data <- "
+    p(x,y,z,w_1,w_2|s_1,s_2)
+    p(z|s_1)
+  "
R> query <- "p(y|do(x))"
R> graph <- "
+    w_1 -> w_2
+    z -> w_2
+    x -> y
+    z -> y
+    z -> s_2
+    w_1 -> x
+    w_2 -> x
+    w_1 -> s_1
+  "
R> dosearch(data, query, graph, selection_bias = "s_1, s_2")

\sum_{z}\left(p(z|s_1)p(y|w_2,x,w_1,z,s_1,s_2)\right)
```

6.4. Systematic analysis of bivariate missing data problems

We apply do-search using the extended rule set of Table 4 for all identifiability problems in bivariate missingness graphs. By bivariate missingness graphs we mean semi-Markovian graphs for two variables, X and Y, and their missingness indicators, R_X and R_Y. Noting that edges from $\{R_X, R_Y\}$ to $\{X, Y\}$ are not allowed, there are 9216 such graphs. We consider only 6144 graphs of which 3072 have the edge $X \to Y$ and 3072 do not have an edge between X and Y. Graphs with the edge $Y \to X$ are obtained from the studied graphs by swapping the roles of X and Y. The maximum number of edges in a bivariate missingness graph is 12 (when a bidirected edge is counted as a single edge).

The available theoretical results for missing data problems include a theorem by Mohan *et al.* (2013) that gives a sufficient and necessary condition for the identifiability of the joint distribution $P(\mathbf{V})$ but is restricted to graphs that do not have edges between the missingness indicators (Row 9 of Table 1). In our example, 5120 graphs out of 6144 have such edges. The algorithm by Shpitser *et al.* (2015) does not have this restriction but it is not complete as shown by Bhattacharya *et al.* (2019) (Row 10 of Table 1). It follows from the results of Bhattacharya *et al.* (2019) that the rules of Table 4 are not complete for missing data problems. Differently from the theorem by Mohan *et al.* (2013) and the algorithms by Shpitser *et al.* (2015) and Bhattacharya *et al.* (2019), do-search can however address missing data problems where we consider identification of a marginal or a conditional distribution. In addition, do-search can address missing data problems with multiple input distributions.



(a) Graphs with arrow $X \to Y$.

(b) Graphs where there is neither $X \to Y$ nor $Y \to X$.

Figure 10: Venn diagrams indicating the number of graphs were different distributions can be identified do-search. The intersection of $P(X)$ and $P(Y \mid X)$ shows the number of graphs were $P(X, Y)$ can be identified. The total number of possible graphs is 3072 in both cases.

The queries $P(X, Y)$, $P(X)$, $P(Y)$, $P(Y \mid X)$ and $P(Y \mid \mathrm{do}(X))$ were evaluated using do-search in these 6144 graphs with the input distribution $P(X^*, Y^*, R_X, R_Y)$. The results are summarized by Venn diagrams in Figure 10. The results are also available as a data set `bivariate_missingness` in the R package **dosearch**. Using this data set we are able to showcase examples on non-identifiability and find interesting special cases by direct evaluation of all possible bivariate missingness graphs. The first example relates non-identifiability to the of the number of edges present in the graph.

Example 1. *Let K denote the number of edges in a bivariate missingness graph that does not have edge $Y \to X$. The joint distribution $P(X, Y)$ is not identifiable by do-search if $K > 5$, marginal distribution $P(X)$ is not identifiable by do-search if $K > 9$, marginal distribution $P(Y)$ and conditional distribution $P(Y \mid X)$ are not identifiable by do-search if $K > 8$.*

The next example specifies the graph with the largest number of edges where both the joint distribution of X and Y and the causal effect of X on Y can be identified.

Example 2. *The graph in Figure 11(a) is the only bivariate missingness graph that (i) has edge $X \to Y$, (ii) has five edges, and (iii) allows for the identification of $P(X, Y)$ and $P(Y \mid do(X))$ by do-search.*

The third example specifies the graph with the largest number of edges where the marginal distributions are identifiable while the joint distribution and the causal effect of X on Y are non-identifiable.

Example 3. *The graph in Figure 11(b) is the only bivariate missingness graph that (i) has five edges, and (ii) allows for the identification of $P(X)$ and $P(Y)$, and (iii) does not allow for the identification of $P(X, Y)$ or $P(Y \mid do(X))$ by do-search. No bivariate missingness graph that has more than five edges fulfills the conditions (ii) and (iii).*

Some interesting examples are shown in Figure 11. Graphs (a) and (b) are the unique graphs that fulfill the conditions specified in Examples 2 and 3, respectively. Graph (c) is the graph

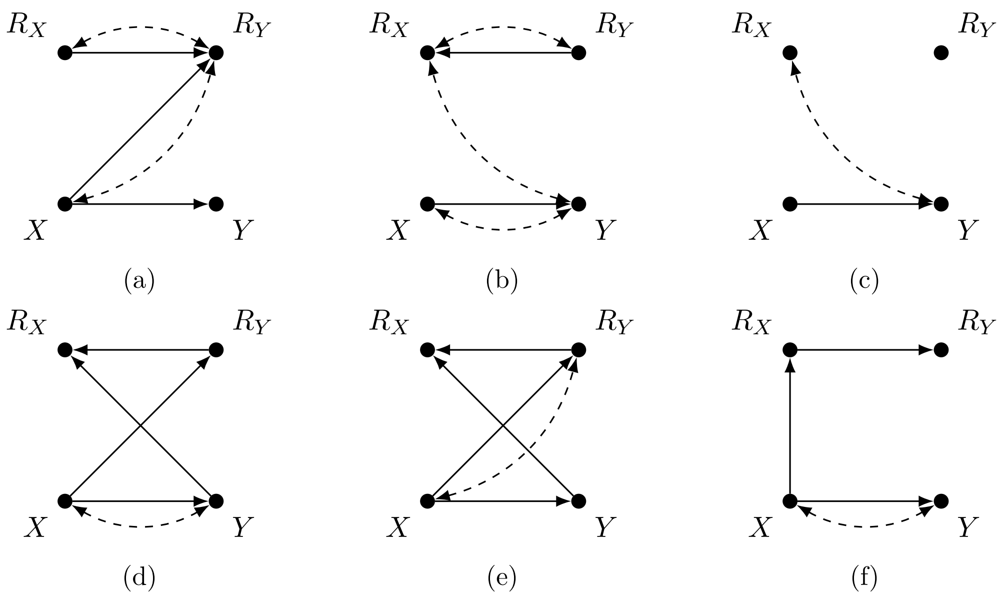


Figure 11: Missingness graphs used as example cases. Proxy variables are omitted for clarity.

with the smallest number of edges where marginals $P(X)$ and $P(Y)$ can be identified but the joint distribution $P(X, Y)$ or causal effect $P(Y \mid \mathrm{do}(X))$ cannot be identified by do-search. In Graph (d), $P(X)$, $P(Y)$, $P(X, Y)$ and $P(Y \mid \mathrm{do}(X))$ are not identifiable by do-search but the conditional distribution $P(Y \mid X)$ can be identified as follows

$$P(Y \mid X) = \frac{P(Y \mid R_Y = 1)P(X \mid Y, R_X = 1, R_Y = 1)}{\sum_{Y'} P(Y' \mid R_Y = 1)P(X \mid Y', R_X = 1, R_Y = 1)}. \tag{3}$$

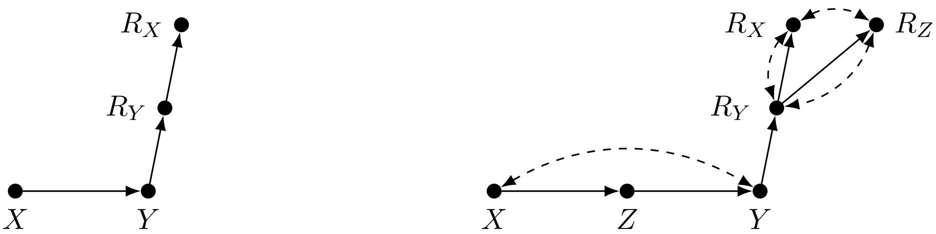
In Equation 3, the numerator resembles the joint distribution $P(X, Y \mid R_X = 1, R_Y = 1)$ but is different because Y and R_X are not independent. The denominator is the marginal of this pseudo joint distribution. In Graph (e), $P(X)$, $P(Y)$ and $P(X, Y)$ are not identifiable by do-search but $P(Y \mid X)$ and $P(Y \mid \mathrm{do}(X))$ are identifiable and can be both estimated with Equation 3. In Graph (f), $P(X, Y)$, $P(X)$ and $P(Y \mid \mathrm{do}(X))$ are not identifiable by do-search but $P(Y)$ and $P(Y \mid X)$ can be identified as follows

$$\begin{aligned} P(Y) &= \sum_{R_X, X^*} P(Y \mid X^*, R_X, R_Y = 1)P(R_X, X^*), \\ P(Y \mid X) &= P(Y \mid X, R_X = 1, R_Y = 1) \end{aligned} \tag{4}$$

In Equation 4, the summation also goes over the cases where $X^* = \mathrm{NA}$ and the distribution of Y must be estimated also on the condition that X is not observed.

6.5. Causal inference under case-control design

Case-control design (Breslow 1996) is commonly used in epidemiology to study risk factors of rare diseases. In the basic setup, a fixed number of disease cases and a fixed number of controls are selected for the risk factor measurements. When the disease is rare, this design leads to substantial savings in the sample size compared to simple random sampling. Figure 12(a) shows the missingness graph for a situation where the inclusion to the study (indicator R_Y)



(a) Basic case-control design.

(b) Case-control design for the front-door situation.

Figure 12: Missingness graph for the case-control examples.

depends on the disease endpoint Y. The risk factors X are measured for the subset $R_Y = 1$ but occasionally the values are missing (indicator R_X). It is immediately seen that neither the causal effect $P(Y \mid \mathrm{do}(X))$ nor conditional distribution $P(Y \mid X)$ can be identified because of the edge $Y \to R_Y$. However, if the prevalence of the disease in the population, i.e., the marginal distribution $P(Y)$, is known, the causal effect $P(Y \mid \mathrm{do}(X))$ can be identified. The result is provided by do-search

$$P(Y \mid \mathrm{do}(X)) = \frac{P(Y)P(X \mid Y, R_Y = 1, R_X = 1)}{\sum_{Y'} P(Y')P(X \mid Y', R_Y = 1, R_X = 1)}. \tag{5}$$

In typical applications response Y is binary but in the non-parametric formula of Equation 5 response can be discrete or continuous. A more complicated example is shown in Figure 12(b) where the causal effect of risk factor X on disease endpoint Y fulfills the front-door criterion (Pearl 1995) with respect to mediator Z and the data are collected from a case-control design where the selection depends Y and there is occasional item non-response in X and Z. We observe data $P(Y^*, X^*, Z^*, R_Y, R_X, R_Z)$ and know the marginal distribution $P(Y)$ from other sources. Applying do-search we obtain the result

$$\begin{aligned} &P(Y \mid \mathrm{do}(X)) = \\ &\quad \sum_Z \left[\frac{\sum_{Y'} P(Y')P(X, Z \mid Y', R_X = 1, R_Y = 1, R_Z = 1)}{\sum_{Z',Y'} P(Y')P(X, Z' \mid Y', R_X = 1, R_Y = 1, R_Z = 1)} \times \right. \\ &\qquad \sum_{X'} \left(\sum_{Y',Z'} P(Y')P(X', Z' \mid Y', R_X = 1, R_Y = 1, R_Z = 1) \times \right. \\ &\qquad\qquad \left. \left. \frac{P(Y)P(X', Z \mid Y, R_X = 1, R_Y = 1, R_Z = 1)}{\sum_{Y'} P(Y')P(X', Z \mid Y', R_X = 1, R_Y = 1, R_Z = 1)} \right) \right]. \end{aligned} \tag{6}$$

Expression 6 follows the general structure of the front-door adjustment

$$P(Y \mid \mathrm{do}(X)) = \sum_Z P(Z \mid X) \sum_{X'} P(X')P(Y \mid X', Z),$$

where

$$P(Z \mid X) = \frac{\sum_{Y'} P(Y')P(X, Z \mid Y', R_X = 1, R_Y = 1, R_Z = 1)}{\sum_{Z',Y'} P(Y')P(X, Z' \mid Y', R_X = 1, R_Y = 1, R_Z = 1)},$$
$$P(X) = \sum_{Y',Z'} P(Y')P(X, Z' \mid Y', R_X = 1, R_Y = 1, R_Z = 1),$$
$$P(Y \mid X, Z) = \frac{P(Y)P(X, Z \mid Y, R_X = 1, R_Y = 1, R_Z = 1)}{\sum_{Y'} P(Y')P(X, Z \mid Y', R_X = 1, R_Y = 1, R_Z = 1)}.$$

Note that $P(X, Y, Z) = P(Y)P(X, Z \mid Y, R_X = 1, R_Y = 1, R_Z = 1)$. In (Karvanen 2015), a similar example was studied assuming that X, Z and Y are binary but in Expression 6 there are no such restrictions. This factorization can be obtained in R as follows

```
R> data <- "
+    p(x*,y*,z*,r_x,r_y,r_z)
+    p(y)
+  "
R> graph <- "
+    x -> z
+    z -> y
+    y -> r_y
+    x <-> y
+    r_y -> r_x
+    r_y -> r_z
+    r_y <-> r_x
+    r_y <-> r_z
+    r_z <-> r_x
+  "
R> md <- "r_x : x, r_y : y, r_z : z"
R> query1 <- "p(z|x)"
R> query2 <- "p(x)"
R> query3 <- "p(y|x,z)"
R> dosearch(data, query1, graph, missing_data = md)

\frac{\sum_{y}\left(p(y)p(x,z|r_x = 1,y,r_y = 1,r_z = 1)\right)}
  {\sum_{z} \sum_{y}\left(p(y)p(x,z|r_x = 1,y,r_y = 1,r_z = 1)\right)}

R> dosearch(data, query2, graph, missing_data = md)

\sum_{y,z}\left(p(y)p(x,z|r_x = 1,y,r_y = 1,r_z = 1)\right)

R> dosearch(data, query3, graph, missing_data = md)

\frac{\left(p(y)p(x,z|r_x = 1,y,r_y = 1,r_z = 1)\right)}
  {\sum_{y} \left(p(y)p(x,z|r_x = 1,y,r_y = 1,r_z = 1)\right)}
```

7. Discussion

The presented algorithm, do-search, removes the need for manual application of do-calculus, which is time-consuming and prone to errors. Systematic analyses such as the one in Section 6.4 are practically unreachable with manual application of do-calculus. Superiority of do-search over a simple forwards breadth-first search was attained through a combination of a search heuristic and a reduction of the search space. Some further approaches were attempted but later discarded as non-beneficial. These include caching separation criteria that hold in the graph after they are first evaluated, pre-computing valid subsets for each subset size and enumerating subsets in an order of increasing cardinality.

As the simulations showed, our intuitive heuristic yielded significant improvements in search performance. The proximity function defined in Section 3.2 uses only the information contained in the distributions themselves. One approach could be to also take the structure of the graph into account in the proximity function. Further study is needed for finding a heuristic that performs well when missing data mechanisms are present in the graph.

The scalability of do-search is limited due to vast search space of possibly identified causal effects. Currently, algorithms with polynomial complexity currently exist only for the simpler problems (see Table 1). However, based on the simulation results, do-search solves identifiability problems in graphs of ten vertices in under two minutes on average. By our observation, graphs typically analyzed in literature related to identifiability problems have fewer vertices. The theoretical computational complexity of the general form of the causal identifiability problem defined in Section 2 remains an important and interesting question.

The search could also be used to obtain formulas that are in some sense simpler than those produced by existing identifiability algorithms. A simplification algorithm by Tikka and Karvanen (2017b) functions as a post-processing step after the identifying formula has already been obtained by the ID algorithm. Given a measure of simplicity, the search heuristic could be adjusted to find simple formulas directly without resorting to separate simplification procedures. In some specific scenarios, such as the standard causal effect identifiability problem, an approach known as pruning (Tikka and Karvanen 2018) could be incorporated into the search. Pruning refers to the removal of vertices from the graph, that are not required for determining identifiability.

Finally we note that identifiability has also been studied under the assumption that the functional relationships depicted by the causal model are linear (Angrist, Imbens, and Rubin 1996; Van der Zander and Liśkiewicz 2016; Chen, Kumor, and Bareinboim 2017) or nonparametric with additive error terms (Peters, Mooij, Janzing, and Schölkopf 2014; Peña and Bendtsen 2017) and when the causal graph is not completely known (Maathuis, Kalisch, and Bühlmann 2009; Entner, Hoyer, and Spirtes 2013; Hyttinen *et al.* 2015; Perković *et al.* 2015; Malinsky and Spirtes 2017; Jaber, Zhang, and Bareinboim 2018). Extending the search in these directions is an interesting line of future research.

8. Conclusion

We presented do-search: a do-calculus based search capable of solving identifiability problems for which no known solutions exist. This contribution is especially useful for researchers working in the field of causal inference to confirm theoretical results or to find counterexamples to identifiability claims. In practical terms, the search can also provide solutions to complicated problems such as combining transportability and selection bias, recovering from multiple bias sources or identifying causal quantities in the presence of missing data that cannot be solved by any other existing method. The R package **dosearch** providing an implementation of do-search is available on CRAN.

Acknowledgments

This work belongs to the thematic research area "Decision analytics utilizing causal models and multiobjective optimization" (DEMO) supported by Academy of Finland (grant number 311877). AH was supported by Academy of Finland through grant 295673. The authors wish to acknowledge CSC – IT Center for Science, Finland, for computational resources.

References

Angrist JD, Imbens GW, Rubin DB (1996). "Identification of Causal Effects Using Instrumental Variables." *Journal of the American Statistical Association*, **91**(434), 444–455. doi:10.2307/2291629.

Bareinboim E, Pearl J (2012a). "Causal Inference by Surrogate Experiments: z-Identifiability." In N de Freitas, K Murphy (eds.), *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence*, pp. 113–120. AUAI Press.

Bareinboim E, Pearl J (2012b). "Controlling Selection Bias in Causal Inference." In ND Lawrence, M Girolami (eds.), *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, volume 22, pp. 100–108.

Bareinboim E, Pearl J (2013). "A General Algorithm for Deciding Transportability of Experimental Results." *Journal of Causal Inference*, **1**, 107–134. doi:10.1515/jci-2012-0004.

Bareinboim E, Pearl J (2014). "Transportability from Multiple Environments with Limited Experiments: Completeness Results." In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pp. 280–288.

Bareinboim E, Tian J (2015). "Recovering Causal Effects from Selection Bias." In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp. 3475–3481.

Bareinboim E, Tian J, Pearl J (2014). "Recovering from Selection Bias in Causal and Statistical Inference." In *Proceedings of the 28th AAAI Conference on Neural Information Processing Systems*.

Bhattacharya R, Nabi R, Shpitser I, Robins JM (2019). "Identification in Missing Data Models Represented by Directed Acyclic Graphs." In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence*.

Breslow NE (1996). "Statistics in Epidemiology: The Case-Control Study." *Journal of the American Statistical Association*, **91**(433), 14–28. doi:10.2307/2291379.

Chen B, Kumor D, Bareinboim E (2017). "Identification and Model Testing in Linear Structural Equation Models Using Auxiliary Variables." In D Precup, YW Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 757–766.

Correa J, Bareinboim E (2017). "Causal Effect Identification by Adjustment under Confounding and Selection Biases." In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*.

Correa J, Tian J, Bareinboim E (2018). "Generalized Adjustment under Confounding and Selection Biases." In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Csardi G, Nepusz T (2006). "The **igraph** Software Package for Complex Network Research." *InterJournal*, **Complex Systems**, 1695.

Danks D, Glymour C, Tillman RE (2009). "Integrating Locally Learned Causal Structures with Overlapping Variables." In *Advances in Neural Information Processing Systems*, pp. 1665–1672.

Dawid AP (2002). "Influence Diagrams for Causal Modelling and Inference." *International Statistical Review*, **70**(2), 161–189. doi:10.2307/1403901.

Eddelbuettel D, François R (2011). "**Rcpp**: Seamless R and C++ Integration." *Journal of Statistical Software*, **40**(8), 1–18. doi:10.18637/jss.v040.i08.

Entner D, Hoyer P, Spirtes P (2013). "Data-Driven Covariate Selection for Nonparametric Estimation of Causal Effects." In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics*, volume 31, pp. 256–264. PMLR.

Greenland S, Robins JM, Pearl J (1999). "Confounding and Collapsibility in Causal Inference." *Statistical Science*, **14**(1), 29–46. doi:10.1214/ss/1009211805.

Huang Y, Valtorta M (2006a). "Identifiability in Causal Bayesian Networks: A Sound and Complete Algorithm." In *Proceedings of the 21st National Conference on Artificial Intelligence – Volume 2*, pp. 1149–1154. AAAI Press.

Huang Y, Valtorta M (2006b). "Pearl's Calculus of Intervention Is Complete." In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, pp. 217–224. AUAI Press.

Hyttinen A, Eberhardt F, Hoyer PO (2012). "Causal Discovery of Linear Cyclic Models from Multiple Experimental Data Sets with Overlapping Variables." In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence*, pp. 387–396.

Hyttinen A, Eberhardt F, Järvisalo M (2015). "Do-Calculus When the True Graph Is Unknown." In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*, pp. 395–404. AUAI Press.

Jaber A, Zhang J, Bareinboim E (2018). "Causal Identification under Markov Equivalence." In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence*, pp. 978–987. AUAI Press.

Kalisch M, Mächler M, Colombo D, Maathuis MH, Bühlmann P (2012). "Causal Inference Using Graphical Models with the R Package **pcalg**." *Journal of Statistical Software*, **47**(11), 1–26. doi:10.18637/jss.v047.i11.

Karvanen J (2015). "Study Design in Causal Models." *Scandinavian Journal of Statistics*, **42**(2), 361–377. doi:10.1111/sjos.12110.

Koller D, Friedman N (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Lauritzen SL (2000). "Causal Inference from Graphical Models." In OE Barndorff-Nielsen, DR Cox, C Klüppelberg (eds.), *Complex Stochastic Systems*, pp. 67–107. Chapman & Hall/CRC. doi:10.1201/9781420035988.

Lee S, Correa J, Bareinboim E (2019). "General Identifiability with Arbitrary Surrogate Experiments." In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence*.

Little RJA, Rubin DB (1986). *Statistical Analysis with Missing Data*. John Wiley & Sons, New York.

Maathuis MH, Colombo D (2015). "A Generalized Backdoor Criterion." *The Annals of Statistics*, pp. 1060–1088. doi:10.1214/14-aos1295.

Maathuis MH, Kalisch M, Bühlmann P (2009). "Estimating High-Dimensional Intervention Effects from Observational Data." *The Annals of Statistics*, **37**(6A), 3133–3164. doi:10.1214/09-aos685.

Malinsky D, Spirtes P (2017). "Estimating Bounds on Causal Effects in High-Dimensional and Possibly Confounded Systems." *International Journal of Approximate Reasoning*, **88**, 371–384. doi:10.1016/j.ijar.2017.06.005.

Mohan K, Pearl J (2018). "Graphical Models for Processing Missing Data." arXiv: 1801.03583 [stat.ME], URL https://arxiv.org/abs/1801.03583.

Mohan K, Pearl J, Tian J (2013). "Graphical Models for Inference with Missing Data." In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pp. 1277–1285.

Pearl J (1995). "Causal Diagrams for Empirical Research." *Biometrika*, **82**(4), 669–688. doi:10.2307/2337329.

Pearl J (2009). *Causality: Models, Reasoning, and Inference*. 2nd edition. Cambridge University Press.

Peña JM, Bendtsen M (2017). "Causal Effect Identification in Acyclic Directed Mixed Graphs and Gated Models." *International Journal of Approximate Reasoning*, **90**, 56–75. doi:10.1016/j.ijar.2017.06.015.

Perković E, Textor J, Kalisch M, Maathuis M (2015). "A Complete Generalized Adjustment Criterion." In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*, pp. 682–691. AUAI Press.

Peters J, Mooij JM, Janzing D, Schölkopf B (2014). "Causal Discovery with Continuous Additive Noise Models." *Journal of Machine Learning Research*, **15**, 2009–2053.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Richardson T (2003). "Markov Properties for Acyclic Directed Mixed Graphs." *Scandinavian Journal of Statistics*, **30**(1), 145–157. doi:10.1111/1467-9469.00323.

Sharma A, Kiciman E (2019). ***DoWhy**: A Python Package for Causal Inference.* URL https://github.com/microsoft/dowhy.

Shpitser I, Mohan K, Pearl J (2015). "Missing Data as a Causal and Probabilistic Problem." In M Meila, T Heskes (eds.), *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*, pp. 802–811. AUAI Press.

Shpitser I, Pearl J (2006a). "Identification of Conditional Interventional Distributions." In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, pp. 437–444. AUAI Press.

Shpitser I, Pearl J (2006b). "Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models." In *Proceedings of the 21st National Conference on Artificial Intelligence – Volume 2*, pp. 1219–1226. AAAI Press.

Shpitser I, Pearl J (2008). "Complete Identification Methods for the Causal Hierarchy." *Journal of Machine Learning Research*, **9**, 1941–1979.

Spirtes P, Glymour C, Scheines R (1993). *Causation, Prediction, and Search*. 2nd edition. Springer-Verlag, New York. doi:10.1007/978-1-4612-2748-9.

Textor J, Van der Zander B, Gilthorpe MS, Liśkiewicz M, Ellison GTH (2016). "Robust Causal Inference Using Directed Acyclic Graphs: The R package **dagitty**." *International Journal of Epidemiology*, **45**(6), 1887–1894. doi:10.1093/ije/dyw341.

Tikka S, Hyttinen A, Karvanen J (2021). ***dosearch**: Causal Effect Identification from Multiple Incomplete Data Sources*. R package version 1.0.8, URL https://CRAN.R-project.org/package=dosearch.

Tikka S, Karvanen J (2017a). "Identifying Causal Effects with the R Package **causaleffect**." *Journal of Statistical Software*, **76**(12), 1–30. doi:10.18637/jss.v076.i12.

Tikka S, Karvanen J (2017b). "Simplifying Probabilistic Expressions in Causal Inference." *Journal of Machine Learning Research*, **18**(36), 1–30.

Tikka S, Karvanen J (2018). "Enhancing Identification of Causal Effects by Pruning." *Journal of Machine Learning Research*, **18**(194), 1–23.

Tikka S, Karvanen J (2019). "Surrogate Outcomes and Transportability." *International Journal of Approximate Reasoning*, **108**, 21–37. doi:10.1016/j.ijar.2019.02.007.

Tillman R, Spirtes P (2011). "Learning Equivalence Classes of Acyclic Models with Latent and Selection Variables from Multiple Datasets with Overlapping Variables." In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pp. 3–15.

Triantafillou S, Tsamardinos I (2015). "Constraint-Based Causal Discovery from Multiple Interventions over Overlapping Variable Sets." *Journal of Machine Learning Research*, **16**, 2147–2205.

Triantafillou S, Tsamardinos I, Tollis I (2010). "Learning Causal Structure from Overlapping Variable Sets." In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pp. 860–867.

Van der Zander B, Liśkiewicz M (2016). "On Searching for Generalized Instrumental Variables." In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*.

Van der Zander B, Liśkiewicz M, Textor J (2019). "Separators and Adjustment Sets in Causal Graphs: Complete Criteria and an Algorithmic Framework." *Artificial Intelligence*, **270**, 1–40. doi:10.1016/j.artint.2018.12.006.

Affiliation:

Santtu Tikka
Department of Mathematics and Statistics
Faculty of Mathematics and Science
University of Jyvaskyla
P.O. Box 35, FI-40014, Finland
E-mail: santtu.tikka@jyu.fi

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
August 2021, Volume 99, Issue 5 *Submitted:* 2019-12-05
doi:10.18637/jss.v099.i05 *Accepted:* 2020-10-26