

Estimating Causal Effects in Partially Directed Parametric Causal Factor Graphs

Malte Luttermann¹, Tanya Braun², Ralf Möller³, and Marcel Gehrke³

¹ German Research Center for Artificial Intelligence (DFKI), Lübeck
`malte.luttermann@dfki.de`

² Data Science Group, University of Münster, Germany
`tanya.braun@uni-muenster.de`

³ Institute for Humanities-Centered Artificial Intelligence, University of Hamburg
`{ralf.moeller,marcel.gehrke}@uni-hamburg.de`

Abstract. Lifting uses a representative of indistinguishable individuals to exploit symmetries in probabilistic relational models, denoted as parametric factor graphs, to speed up inference while maintaining exact answers. In this paper, we show how lifting can be applied to causal inference in partially directed graphs, i.e., graphs that contain both directed and undirected edges to represent causal relationships between random variables. We present partially directed parametric causal factor graphs (PPCFGs) as a generalisation of previously introduced parametric causal factor graphs, which require a fully directed graph. We further show how causal inference can be performed on a lifted level in PPCFGs, thereby extending the applicability of lifted causal inference to a broader range of models requiring less prior knowledge about causal relationships.

Keywords: causal models; probabilistic relational models; lifted inference.

1 Introduction

A fundamental problem for an intelligent agent performing reasoning under uncertainty is to compute the effect of an action on a certain random variable (randvar) on other randvars. When computing the effect of an action on a specific randvar, it is crucial to deploy the semantics of an intervention instead of performing a classical conditioning on that randvar [22, Chapter 4]. An intervention acting on a randvar R can be thought of as setting R to a fixed value and removing all incoming influences on the value of R . In practice, generally not all causal relationships in a given model are known and thus, only a partially directed graphical model is available. In such a partially directed graph, directed edges represent cause-effect relationships and undirected edges represent causal relationships whose direction is unknown. In this paper, we solve the problem of efficiently estimating causal effects of actions in partially directed lifted probabilistic models, denoted as parametric factor graphs. Lifted representations are not only more expressive than propositional models such as factor graphs, but also allow for tractable probabilistic inference with respect to domain sizes of logical variables (logvars) by exploiting symmetries.

Previous Work. The estimation of causal effects using causal graphical models in form of directed acyclic graphs in combination with observational data has been extensively studied in the literature (see, e.g., [22,23,27]). Some works incorporate causal knowledge into (propositional) factor graphs (which are originally undirected graphical models) to enable the estimation of causal effects in factor graphs [6,29]. In practice, the underlying causal graph is often not fully known and hence, identifying and estimating causal effects when provided with observational data and a partially directed graph has been investigated [7,10,16,24]. However, all of these works perform causal effect estimation on a propositional level and thus lack the expressivity of relational logic, for example to capture the relationships between individual objects. To represent individual objects and the relationships between them, Poole [25] introduces parametric factor graphs as lifted representations, which combine relational logic and probabilistic models, thereby allowing to encode that certain properties hold for groups of indistinguishable objects. In probabilistic inference, lifting exploits symmetries to speed up inference while maintaining exact answers [20]. Over the past years, both algorithms for symmetry detection [1,8,11,13,14,15] allowing the construction of lifted representations such as parametric factor graphs as well as various lifted inference algorithms operating on parametric factor graphs have been developed and further refined [2,3,4,5,9,18,25,28]. More recently, Luttermann et al. [12] introduce parametric causal factor graphs (PCFGs) as an extension of parametric factor graphs allowing to incorporate causal knowledge into a lifted representation. Nevertheless, the authors assume that the causal relationships between the involved randvars are fully known, which is rarely the case in practical settings.

Our Contributions. We introduce partially directed parametric causal factor graphs (PPCFGs) as a generalisation of PCFGs to obtain a formalism that compactly encodes a full joint distribution over a set of randvars and at the same time incorporates causal knowledge in the model, if available. The major advantage of a PPCFG over an PCFG is that not all causal relationships between the involved randvars need to be known, thereby reducing the amount of prior knowledge required and thus making the model more suitable for many practical settings. We further define d -separation in PPCFGs to reason about conditional independence. In addition to that, we present an algorithm to efficiently estimate causal effects in PPCFGs on a lifted level, i.e., a representative of indistinguishable objects is used for computations to speed up inference. Our algorithm identifies whether a causal effect can be uniquely determined from the given PPCFG and if so, outputs the causal effect. If the undirected edges in the PPCFG lead to a causal effect being not uniquely identifiable, our algorithm efficiently enumerates all possible causal effects while operating on a lifted level.

Structure of this Paper. We begin by introducing necessary background information and notations. Afterwards, we present PPCFGs as a generalisation of PCFGs, allowing both for directed and undirected edges in the model and then define d -separation in PPCFGs. Thereafter, we provide an algorithm to efficiently estimate causal effects in PPCFGs before we conclude.

2 Preliminaries

We begin to introduce parameterised randvars (PRVs), which use logvars as parameters to represent sets of indistinguishable randvars.

Definition 1 (Parameterised Random Variable). *Let \mathbf{R} be a set of randvar names, \mathbf{L} a set of logvar names, and \mathbf{D} a set of constants. All sets are finite. Each logvar L has a domain $\text{dom}(L) \subseteq \mathbf{D}$. A constraint is a tuple $(\mathcal{X}, C_{\mathcal{X}})$ of a sequence of logvars $\mathcal{X} = (X_1, \dots, X_n)$ and a set $C_{\mathcal{X}} \subseteq \times_{i=1}^n \text{dom}(X_i)$. The symbol \top for C marks that no restrictions apply, i.e., $C_{\mathcal{X}} = \times_{i=1}^n \text{dom}(X_i)$. A PRV $R(L_1, \dots, L_n)$, $n \geq 0$, is a syntactical construct of a randvar $R \in \mathbf{R}$ possibly combined with logvars $L_1, \dots, L_n \in \mathbf{L}$ to represent a set of randvars. If $n = 0$, the PRV is parameterless and forms a propositional randvar. A PRV A (or logvar L) under constraint C is given by $A|_C$ ($L|_C$), respectively. We may omit $|\top$ in $A|_{\top}$ or $L|_{\top}$. The term $\text{range}(A)$ denotes the possible values of a PRV A . An event $A = a$ denotes the occurrence of PRV A with range value $a \in \text{range}(A)$.*

Example 1. Consider $\mathbf{R} = \{Comp, Sal, Rev\}$ for competence, salary, and revenue, respectively, and $\mathbf{L} = \{E\}$ with $\text{dom}(E) = \{alice, bob, charlie\}$ (employees), combined into PRVs $Comp(E)$, $Sal(E)$, and Rev with $\text{range}(Comp(E)) = \text{range}(Sal(E)) = \text{range}(Rev) = \{\text{low}, \text{medium}, \text{high}\}$.

A parametric factor (parfactor) describes a function, mapping argument values to positive real numbers, of which at least one is non-zero.

Definition 2 (Parfactor). *Let Φ denote a set of factor names. We denote a parfactor g by $\phi(\mathcal{A})|_C$ with $\mathcal{A} = (A_1, \dots, A_n)$ being a sequence of PRVs, $\phi: \times_{i=1}^n \text{range}(A_i) \mapsto \mathbb{R}^+$ being a function with name $\phi \in \Phi$ mapping argument values to a positive real number called potential, and C being a constraint on the logvars of \mathcal{A} . We may omit $|\top$ in $\phi(\mathcal{A})|_{\top}$. The term $lv(Y)$ refers to the logvars in some element Y , a PRV, a parfactor, or sets thereof. The term $gr(Y|_C)$ denotes the set of all instances (groundings) of Y with respect to constraint C .*

Example 2. Take a look at the parfactor $g_1 = \phi_1(Comp(E), Rev)|_{\top}$. Assuming the same ranges of the PRVs and the same domains of the logvars as in Ex. 1, g_1 specifies $|\text{range}(Comp(E))| \cdot |\text{range}(Rev)| = 9$ input-output mappings $\phi_1(\text{low}, \text{low}) = \varphi_1$, $\phi_1(\text{low}, \text{medium}) = \varphi_2$, $\phi_1(\text{low}, \text{high}) = \varphi_3$, and so on with $\varphi_i \in \mathbb{R}^+$ for all $i = 1, \dots, 9$. Further, $lv(g_1) = \{E\}$ and $gr(g_1|_{\top}) = \{\phi_1(Comp(alice), Rev), \phi_1(Comp(bob), Rev), \phi_1(Comp(charlie), Rev)\}$.

A PCFG is then built from a set of parfactors $\{g_1, \dots, g_m\}$ and the causal relationships between the PRVs, which are encoded by the direction of the edges in the graph structure of the PCFG [12]. In its original form, a PCFG is a fully directed graph, but we extend this definition to allow for both directed and undirected edges in the following section.

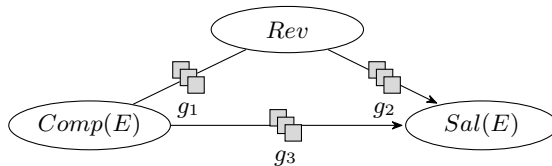


Fig. 1: A PPCFG modelling the interplay of a company’s revenue and its employees’ competences and salaries (without input-output mappings of parfactors).

3 Partially Directed Parametric Causal Factor Graphs

We now move on to define PPCFGs as lifted models that are able to incorporate partial causal knowledge, thereby allowing to exploit symmetries (in form of indistinguishable individuals) to speed up both probabilistic and causal inference by performing lifted inference.

Definition 3 (Partially Directed Parametric Causal Factor Graph). A PPCFG is a graph $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ that consists of variable nodes \mathbf{A} , factor nodes \mathbf{G} ($\mathbf{A} \cap \mathbf{G} = \emptyset$), and a set of edges \mathbf{E} . Each variable node $A \in \mathbf{A}$ represents a PRV A and every factor node $g \in \mathbf{G}$ represents a parfactor $g = \phi(\mathcal{A})|_C$, where $\mathcal{A} = (A_1, \dots, A_k)$ with $A_1 \in \mathbf{A}, \dots, A_k \in \mathbf{A}$ is a sequence of PRVs, $\phi: \times_{i=1}^k \text{range}(A_i) \mapsto \mathbb{R}^+$ is a function, and C is a constraint on the logvars of \mathcal{A} . We again may omit $|_T$ in $\phi(\mathcal{A})|_T$. For a variable node $A \in \mathbf{A}$ and a factor node $g \in \mathbf{G}$, there is an undirected edge $g - A \in \mathbf{E}$ if A appears in the argument list of $g = \phi(\mathcal{A})$ and no information about the causal relationships between the PRVs in \mathcal{A} is available. If it is known that all $A' \in \mathcal{A} \setminus \{A\}$ are causes of $A \in \mathcal{A}$ (or if $\mathcal{A} \setminus \{A\} = \emptyset$), the edge $g - A$ can be replaced by a directed edge $g \rightarrow A$.

The semantics of M is given by grounding and building a full joint distribution. With Z as the normalisation constant and \mathcal{A}_k denoting the PRVs appearing in the argument list of $\phi_k(\mathcal{A}_k)$, M represents the full joint distribution

$$P_M = \frac{1}{Z} \prod_{g \in \mathbf{G}} \prod_{\phi_k \in \text{gr}(g)} \phi_k(\mathcal{A}_k).$$

Example 3. Following Exs. 1 and 2, Fig. 1 depicts a PPCFG M modelling the interplay of a company’s revenue and its employees’ competences and salaries. Each parfactor represents a group of ground factors and thus, grounding M results in a partially directed factor graph (see Fig. 2). In this particular example, the causal relationships encoded by M tell us that the revenue of the company influences the salary of each individual employee and the competence of a specific employee influences their salary. Moreover, there is a dependency between the competence of each individual employee and the revenue of the company, but the causal direction is not encoded in M . We humans expect that the competence of the employees influences the revenue of the company, but an autonomous agent might not have this information available, resulting in a partially directed graph.

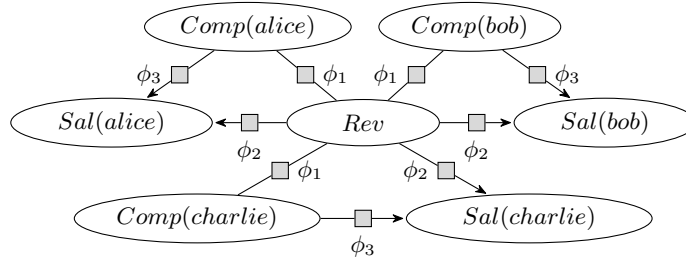


Fig. 2: A visualisation of the resulting model when grounding the PPCFG M given in Fig. 1, where $\text{dom}(E) = \{alice, bob, charlie\}$.

In accordance with the literature, in this paper we consider PPCFGs that do not contain directed cycles. Throughout this paper, we denote the parents of a PRV A in a PPCFG $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ by $\text{Pa}(A, M) = \{A' \in \mathbf{A} \mid \exists g \in \mathbf{G}: (g - A') \in \mathbf{E} \wedge (g \rightarrow A) \in \mathbf{E}\}$. Analogously, we define the children of A in M as $\text{Ch}(A, M) = \{A' \in \mathbf{A} \mid \exists g \in \mathbf{G}: (g \rightarrow A') \in \mathbf{E} \wedge (g - A) \in \mathbf{E}\}$ and the neighbours of A in M as $\text{Ne}(A, M) = \{A' \in \mathbf{A} \mid \exists g \in \mathbf{G}: (g - A') \in \mathbf{E} \wedge (g - A) \in \mathbf{E}\}$.

Note that the semantics of a PPCFG is defined with respect to the set of parfactors and thus is independent of the given causal relationships. A PPCFG hence simultaneously encodes a full joint probability distribution over a set of randvars (represented by PRVs) and a set of causal relationships between those randvars¹. Another important remark is that grounding a PPCFG results in a partially directed factor graph as introduced by Frey [6]. The important advantage is that the PPCFG combines relational logic with probabilistic models, thereby being more expressive than a factor graph and allowing us to exploit symmetries (in form of indistinguishable individuals) to speed up inference.

Example 4. Take a look at Fig. 2, which presents the resulting model when grounding the PPCFG M given in Fig. 1. The model $gr(M)$ now contains the randvars $Comp(alice)$, $Comp(bob)$, $Comp(charlie)$, which are represented by a single PRV $Comp(E)$ with a logvar E , $\text{dom}(E) = \{alice, bob, charlie\}$, in M (analogously for $Sal(alice)$, $Sal(bob)$, $Sal(charlie)$ and $Sal(E)$). Moreover, each parfactor g_i in M represents a set of three (identical) ground factors ϕ_i in the grounded model $gr(M)$. In consequence, the lifted representation M is more compact than its grounded counterpart $gr(M)$, thereby reducing run times required to perform inference. The underlying assumption is that there are indistinguishable objects, here employees, which can be represented by a representative.

Since a PPCFG encodes a full joint probability distribution, a PPCFG can be used to compute marginal distributions of grounded PRVs given observations of specific events. A query asks for a probability distribution (or a specific probability) given a set of observed events or, in the case of an interventional query, given a set of interventions fixing the values of certain (grounded) PRVs.

¹ Thus, the setting of having an PPCFG is equivalent to having a causal graph and observational data for the randvars occurring in the causal graph [22].

Definition 4 (Query). A query $P(Q \mid E_1 = e_1, \dots, E_k = e_k)$ consists of a query term Q and a set of events $\{E_j = e_j\}_{j=1}^k$ with Q and E_j being grounded or parameterless PRVs. To query a specific probability instead of a probability distribution, the query term is an event $Q = q$. An interventional query $P(Q \mid do(R_1 = r_1, \dots, R_k = r_k))$ asks for a probability distribution given that the grounded or parameterless PRVs R_j are set to a fixed value r_j .

Example 5. The query $P(Rev \mid Comp(alice) = \text{high})$ asks for the probability distribution of Rev given that the event $Comp(alice) = \text{high}$ is observed. Keep in mind that, in general, observing and intervening are two different things, e.g., generally $P(Rev \mid Comp(alice) = \text{high}) \neq P(Rev \mid do(Comp(alice) = \text{high}))$.

Before we consider interventions in detail, we first take a closer look at conditional independence in PPCFGs, which is needed to reason about interventions.

4 Conditional Independence in PPCFGs

The notion of d -separation [21] provides a graphical criterion to test for conditional independence in directed acyclic graphs and is essential to compute the effect of an intervention in the sense that all non-causal paths, so-called backdoor paths, need to be blocked to remove spurious effects. Frey [6] extends the notion of d -separation to partially directed factor graphs and we build on this definition to define d -separation in PPCFGs (analogously to d -separation in PCFGs [12]). Note that the semantics of d -separation in PPCFGs is defined on a ground level.

Definition 5 (d -separation). Let $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ be a PPCFG. Given three disjoint sets of randvars \mathbf{X} , \mathbf{Y} , and \mathbf{Z} (subsets of $gr(\mathbf{A})$), we say that \mathbf{X} and \mathbf{Y} are conditionally independent given \mathbf{Z} , written as $\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z}$, if the nodes in \mathbf{Z} block all paths between the nodes in \mathbf{X} and the nodes in \mathbf{Y} in $gr(M)$. A path is a connected sequence of edges (independent of their directions) and it is therefore also possible for a path to pass from a parent of a factor to another parent of that factor. A path is blocked by the nodes in \mathbf{Z} if

1. the path contains the pattern $\phi_1 \rightarrow A \leftarrow \phi_2$ such that neither A nor any of its descendants are in \mathbf{Z} , or
2. the path passes from ϕ_1 through A to ϕ_2 such that it does not contain the pattern $\phi_1 \rightarrow A \leftarrow \phi_2$ and A is in \mathbf{Z} , or
3. the path passes from a parent of a factor ϕ to another parent of ϕ , and neither the child of ϕ nor any of its descendants are in \mathbf{Z} .

Example 6. Consider the grounded PPCFG M depicted in Fig. 2. M encodes, for example, that the competence of *alice* is independent of *bob*'s salary given the revenue of the company, written as $\{Comp(alice)\} \perp\!\!\!\perp \{Sal(bob)\} \mid \{Rev\}$, because all paths from $Comp(alice)$ to $Sal(bob)$ pass through Rev .

Remark 1. Definition 5 is slightly different from the definition of d -separation provided by Frey [6] for partially directed factor graphs in the sense that Def. 5 is more flexible by allowing for variable nodes to have multiple parent factor nodes (e.g., in Fig. 1, $Sal(E)$ has two parent factor nodes g_2 and g_3 instead of a single parent factor node with two inputs from $Comp(E)$ and Rev).

Remark 2. In case a PPCFG M is constructed by hand (instead of being learned from observational data), it is possible to construct a mismatch of the conditional independence statements encoded in the graph structure of M and the conditional independence statements implied by the full joint probability distribution encoded by M . We therefore assume that the graph structure of a given PPCFG M encodes exactly the same conditional independence statements as the underlying full joint probability distribution encoded by M .

As we have seen, the semantics of d -separation in PPCFGs is defined on a ground level. However, it is possible to check for d -separation on a lifted level without having to ground the entire PPCFG. For this purpose, the well-known Bayes-Ball algorithm [26] that allows us to check for conditional independence can be applied on a lifted level, that is, it can be run on the PPCFG instead of on its grounded counterpart [17]. The idea of the Bayes-Ball algorithm is that a bouncing ball is sent through the graph structure such that the ball can pass through a node, bounce back, or be blocked to determine blocked paths (more details can be found in [26]). It is also possible to check whether PRVs (instead of ground randvars) are conditionally independent in a highly efficient manner on a lifted level. In a PPCFG, each PRV A is represented by a single variable node and thus, checking for conditional independence statements that involve A can be done by looking at this single variable node instead of taking into account all groundings $gr(A)$ of A . In contrast, in a propositional setting (i.e., in a ground model), each ground randvar in $gr(A)$ must be looked at individually (for example, to check whether $\{Comp(E)\} \perp\!\!\!\perp \{Sal(E)\} \mid \{Rev\}$ holds, only three variable nodes are of relevance in the lifted representation while $2 \cdot |\text{dom}(E)| + 1$ nodes are of relevance in the ground model).

We next show how the estimation of causal effects, which rely on the notion of an intervention, can efficiently be realised in PPCFGs. The idea is again to avoid grounding the entire PPCFG, if possible.

5 Efficient Estimation of Causal Effects in PPCFGs

To compute the effect of actions carried out on randvars on other randvars, we have to answer interventional queries [22]. An intervention on a grounded or parameterless PRV R , denoted as $do(R = r)$ where $r \in \text{range}(R)$, changes the structure of the underlying model by setting the value of R to r and removing all incoming influences on R . An intervention is defined on a fully directed graph.

Definition 6 (Intervention). *Let $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ be a fully directed PPCFG and let $gr(\mathbf{A}) = \{R_1, \dots, R_n\}$ denote the set of randvars obtained by grounding the PRVs in \mathbf{A} . Any probability distribution entailing the conditional independence statements encoded by M can be factorised as $P(R_1, \dots, R_n) =$*

$\prod_{i=1}^n P(R_i \mid \text{Pa}(R_i, \text{gr}(M)))$. An intervention $do(R'_1 = r'_1, \dots, R'_k = r'_k)$ changes the underlying probability distribution such that

$$P(R_1 = r_1, \dots, R_n = r_n \mid do(R'_1 = r'_1, \dots, R'_k = r'_k)) = \begin{cases} \prod_{R_i \in \{R_1, \dots, R_n\} \setminus \{R'_1, \dots, R'_k\}} P(r_i \mid \text{pa}(R_i, \text{gr}(M))) & \text{if } \forall j \in \{1, \dots, k\}: r_j = r'_j \\ 0 & \text{otherwise,} \end{cases}$$

where $\text{pa}(R_i, \text{gr}(M))$ denotes the values of $\text{Pa}(R_i, \text{gr}(M))$.

Observe that the definition of an intervention refers to a fully directed PPCFG. In general, we deal with PPCFGs that might contain undirected edges and thus represent a whole class of fully directed PPCFGs, namely all fully directed PPCFGs that entail the same conditional independence statements as the initial PPCFG². To determine the semantics of an intervention, we thus need to know the parents of the PRVs on which we intervene. As we deal with PPCFGs that might contain undirected edges, however, we do not always know the real parents of a PRV.

Example 7. Given the PPCFG depicted in Fig. 1, the company might wonder whether it is worth it to send its employee *alice* to an expensive training course to increase the revenue of the company due to the increased competence of *alice*. Hence, the company is interested in computing the quantity $P(\text{Rev} \mid do(\text{Comp}(\text{alice}) = \text{high}))$. Since an intervention can be thought of as setting the value of $\text{Comp}(\text{alice})$ to high and removing all incoming edges of $\text{Comp}(\text{alice})$ in the graph, there are two possible scenarios when computing $P(\text{Rev} \mid do(\text{Comp}(\text{alice}) = \text{high}))$: (i) Rev is a parent of $\text{Comp}(\text{alice})$ in the true model and thus, the underlying probability distribution is changed, or (ii) $\text{Comp}(\text{alice})$ is a parent of Rev in the true model and therefore, the underlying probability distribution remains unchanged as $\text{Comp}(\text{alice})$ has no parents. Without further background information, we do not know which of these two scenarios is actually correct and thus, we need to consider both possibilities.

Fortunately, we do not always have to consider all possible fully directed PPCFGs represented by a given PPCFG when computing the effect of an intervention because there are settings in which the represented fully directed PPCFGs all yield the same effect of the intervention. In particular, in case all parents of the randvars on which we intervene are known, we can uniquely determine the effect of the intervention even if there are still undirected edges in the PPCFG³.

Theorem 1. *Let $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ denote a PPCFG and let $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ be an interventional query with $Q \in \text{gr}(\mathbf{A})$, $R'_1 \in \text{gr}(\mathbf{A}), \dots, R'_k \in \text{gr}(\mathbf{A})$, and $\{Q\} \cap \{R'_1, \dots, R'_k\} = \emptyset$. If it holds that $\text{Ne}(R'_1, \text{gr}(M)) = \emptyset, \dots, \text{Ne}(R'_k, \text{gr}(M)) = \emptyset$, then the result of $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ is identical for all fully directed PPCFGs represented by M .*

² For classical (propositional) directed acyclic graphs, the set of fully directed acyclic graphs representing identical conditional independence statements is known under the name of a *Markov equivalence class*.

³ In fact, this has been shown for propositional models [16,19] and we now transfer this result to relational models.

Proof. Let $\{R_1, \dots, R_\ell\} = gr(\mathbf{A}) \setminus \{Q, R'_1, \dots, R'_k\}$ denote the set of all grounded randvars that do not occur in the given query. From Def. 6 we know that

$$\begin{aligned} & P(Q = q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k)) \\ = & \sum_{r_1 \in \text{range}(R_1)} \dots \sum_{r_\ell \in \text{range}(R_\ell)} \prod_{R_i \in gr(\mathbf{A}) \setminus \{R'_1, \dots, R'_k\}} P(r_i \mid \text{pa}(R_i, gr(M))). \end{aligned} \quad (1)$$

If it holds that $\text{Ne}(R'_1, gr(M)) = \emptyset, \dots, \text{Ne}(R'_k, gr(M)) = \emptyset$, then the parents of R'_1, \dots, R'_k are fully known and identical in all fully directed PPCFGs represented by M . Hence, it remains to be shown that the factorisation of all ground randvars that are not in $\{R'_1, \dots, R'_k\}$ is equivalent for all fully directed PPCFGs represented by M . We know that all fully directed PPCFGs represented by M entail exactly the same conditional independence statements as M and thus, the factorisations induced by these fully directed PPCFGs entail equivalent semantics (just as all Bayesian network structures over a fixed set of randvars entailing the same conditional independence statements induce equivalent factorisations of the underlying probability distribution). Consequently, Eq. (1) yields the same result for all fully directed PPCFGs represented by M .

Theorem 1 implies that we do not have to consider all possible edge directions of the undirected edges in a PPCFG when computing the effect of an intervention but just the possible directions of the undirected edges that are relevant for the intervention, that is, the directions of the undirected edges that are connected to the randvars on which we intervene. Note that all terms required to answer the interventional query can be computed by querying the PPCFG, as the PPCFG compactly encodes the full joint probability distribution over all ground randvars. The semantics of the PPCFG is well-defined even if there are undirected edges in the graph because the factors do not necessarily encode distributions conditioned on the parents but instead encode arbitrary local distributions that factorise the full joint probability distribution.

Intuitively, it becomes clear that the effect of an intervention cannot be uniquely determined if there are undirected edges connected to the randvars on which we intervene because there are different possible parent sets that might result in multiple disjoint effects of the intervention.

Theorem 2. *Let $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ denote a PPCFG and let $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ be an interventional query with $Q \in gr(\mathbf{A})$, $R'_1 \in gr(\mathbf{A})$, $\dots, R'_k \in gr(\mathbf{A})$, and $\{Q\} \cap \{R'_1, \dots, R'_k\} = \emptyset$. If there exists a randvar $R'_i \in \{R'_1, \dots, R'_k\}$ such that $\text{Ne}(R'_i, gr(M)) \neq \emptyset$, then the result of $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ is not guaranteed to be uniquely determined.*

Proof. If there exists a randvar $R'_i \in \{R'_1, \dots, R'_k\}$ such that $\text{Ne}(R'_i, gr(M)) \neq \emptyset$ holds, there are undirected edges connected to R'_i , implying that the parents of R'_i are not guaranteed to be identical in all fully directed PPCFGs represented by M . Since we know that Eq. (1) gives us the result of the given query, the factors being removed from the product on the right hand side of the equation might differ depending on the actual parents of the ground randvars on which we intervene, thereby possibly yielding different results for the query.

Algorithm 1: Lifted Causal Inference in PPCFGs

Input : A PPCFG $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ and an interventional query $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ with $Q \in gr(\mathbf{A})$, $R'_1 \in gr(\mathbf{A}), \dots, R'_k \in gr(\mathbf{A})$ and $\{Q\} \cap \{R'_1, \dots, R'_k\} = \emptyset$.

Output: The set of all possible post-intervention distributions $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$ represented by M .

- 1 $\mathbf{P} \leftarrow \emptyset$;
- 2 $\{R_1, \dots, R_\ell\} \leftarrow gr(\mathbf{A}) \setminus \{Q, R'_1, \dots, R'_k\}$;
- 3 $M' \leftarrow$ PCFG after splitting parfactors in M based on each $R'_i \in \{R'_1, \dots, R'_k\}$;
- 4 **foreach** $\mathbf{C}_1 \subseteq Ne(R'_1, M'), \dots, \mathbf{C}_k \subseteq Ne(R'_k, M')$ *s.t.* $\mathbf{C}_1, \dots, \mathbf{C}_k$ are cliques **do**
- 5 $M'' \leftarrow$ PCFG after orienting all edges $C - \phi - R'_i$ as $C - \phi \rightarrow R'_i$ in M' for all $C \in \mathbf{C}_i$ for every $i \in \{1, \dots, k\}$;
- 6 $M'' \leftarrow$ Arbitrary fully directed PPCFG represented by M'' ;
- 7 **if** *no such M'' exists* **then**
- 8 **continue**;
- 9 $D \leftarrow \sum_{r_1 \in range(R_1)} \dots \sum_{r_\ell \in range(R_\ell)} \prod_{R_i \in gr(\mathbf{A}) \setminus \{R'_1, \dots, R'_k\}} P(r_i \mid pa(R_i, M''))$;
- 10 Add D to \mathbf{P} ;
- 11 **return** \mathbf{P}

Remark 3. There are scenarios in which it is possible to uniquely determine the result of an interventional query even if there are undirected edges connected to the randvars on which we intervene as not all undirected edges can be oriented in both directions (because they are not allowed to introduce a cycle or to change the conditional independence statements encoded in the graph structure)⁴.

Combining the insights from Thms. 1 and 2 naturally leads to an algorithm to compute the effect of interventions in PPCFGs. The idea is that all possible parent sets of the intervention variables have to be considered. If there is just one possible set of parents, the effect of the intervention can be uniquely determined, otherwise there are multiple possible effects that are enumerated. This idea is incorporated in the IDA algorithm and its variants [7,10,16] for single interventions (i.e., interventions $do(R' = r')$) in propositional models. Algorithm 1 displays our proposed algorithm, which extends the idea of just considering the possible parent sets of intervention variables to joint interventions (i.e., interventions $do(R'_1 = r'_1, \dots, R'_k = r'_k)$ where $k \geq 1$) in relational models⁵.

Given a PPCFG $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ and an interventional query $P(Q \mid do(R'_1 = r'_1, \dots, R'_k = r'_k))$, Alg. 1 proceeds as follows to compute the set of all possible results for the given query. First, Alg. 1 splits the parfactors in M based on

⁴ An orientation of an undirected edge alters the conditional independence statements if the new orientation introduces the pattern $R_1 - \phi_1 \rightarrow R_2 \leftarrow \phi_2 - R_3$ such that R_1 and R_3 are not directly connected via a factor—see Item 1 in Def. 5.

⁵ Algorithm 1 transfers the results for joint interventions in propositional models with undirected edges [19] to relational models.

each $R'_i \in \{R'_1, \dots, R'_k\}$. Splitting the parfactors results in a modified PPCFG M' entailing equivalent semantics as M and works as follows [4]. Recall that $R'_i = A(L_1 = l_1, \dots, L_j = l_j)$, $l_1 \in \text{dom}(L_1), \dots, l_j \in \text{dom}(L_j)$, is a specific instance of a PRV $A(L_1, \dots, L_j)$. The idea behind the splitting procedure is that we would like to separate $gr(A)$ into two sets $gr(A) \setminus \{R'_i\}$ and $\{R'_i\}$, as R'_i has to be treated differently than the remaining instances of A due to the intervention. Thus, every parfactor g for which there is an instance $\phi \in gr(g)$ such that R'_i appears in the argument list of ϕ is split. Formally, splitting a parfactor g replaces g by two parfactors $g'_{C'}$ and $g''_{C''}$ and adapts the constraints of $g'_{C'}$ and $g''_{C''}$ such that the inputs of $g'_{C'}$ are restricted to all sequences that contain R'_i and the inputs of $g''_{C''}$ are restricted to the remaining input sequences. After the splitting procedure, the semantics of the model remains unchanged as the groundings of M' are still the same as the groundings of the initial model M —they are just arranged differently across the sets of ground instances.

Algorithm 1 then iterates over all possible parent sets (i.e., over all subsets of undirected neighbours) of R'_1, \dots, R'_k . When considering the subsets of undirected neighbours, it is necessary that all subsets are jointly valid, that is, they are not allowed to alter the conditional independence statements encoded by the model and they must not introduce any cycles when oriented towards R'_1, \dots, R'_k . To ensure the validity of these subsets, they are required to form a clique. A clique C is a subset of nodes such that all pairs of nodes in C are directly connected via a factor⁶. By ensuring that the subsets of undirected neighbours form cliques, the orientation of the undirected neighbours of R'_1, \dots, R'_k towards R'_1, \dots, R'_k does not introduce any pattern $R_1 - \phi_1 \rightarrow R_2 \leftarrow \phi_2 - R_3$ such that R_1 and R_3 are not directly connected via a factor, as due to the clique property R_1 and R_3 are always guaranteed to be directly connected via a factor. Thus, the conditional independence statements encoded by M'' are equivalent to those encoded by M . Having obtained the parent sets of R'_1, \dots, R'_k , Alg. 1 next extends the modified model M'' to any fully directed PPCFG represented by M'' , if such a fully directed PPCFG exists. Afterwards, M'' is guaranteed to be a fully directed PPCFG represented by M and hence, the result of the provided query is given by Eq. (1), which requires us to know the parents of all non-intervention randvars to compute a product over conditional probability distributions. Since M encodes the full joint probability distribution over $gr(\mathbf{A})$, the conditional probability distributions are obtained by querying M , which allows us to compute the results for these queries using lifted probabilistic inference, thereby avoiding to ground the entire model if possible. The resulting post-intervention distribution is then added to the result set \mathbf{P} and the algorithm continues the above steps for the next possible parent set until all possible parent sets have been taken into account.

⁶ Formally, a clique C in a PPCFG $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ is a subset of nodes such that for each pair of nodes $C_1 \in C$, $C_2 \in C$ with $C_1 \neq C_2$ it holds that there exists a factor ϕ such that there is an edge between C_1 and ϕ as well as an edge between C_2 and ϕ in \mathbf{E} (either directed or undirected).

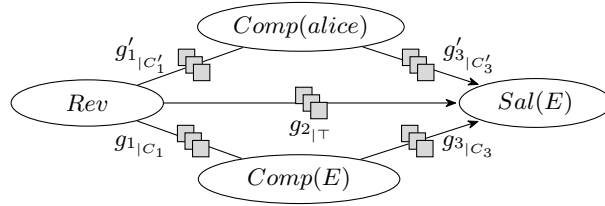


Fig. 3: The modified PPCFG obtained after splitting the parafactors in the PPCFG shown in Fig. 1 to separate $Comp(alice)$ from $Comp(E)$.

Example 8. Assume we want to compute $P(Rev \mid do(Comp(alice) = high))$ in the PPCFG M depicted in Fig. 1. As the intervention operates on $Comp(alice)$, first the parafactors in M are split on $Comp(alice)$ to obtain the modified PPCFG M' shown in Fig. 3. In M' , $Comp(alice)$ is now a separate node in the graph, so its possible parents can be determined. Formally, g_1 has been replaced by two parafactors $g_{1|C_1}$ and $g'_{1|C'_1}$ with constraints $C_1 = (E, \{bob, charlie\})$ and $C'_1 = (E, \{alice\})$. Analogously, g_3 has been replaced by $g_{3|C_3}$ and $g'_{3|C'_3}$. Next, Alg. 1 considers all possible parent sets of $Comp(alice)$, which are given by $Pa_1(Comp(alice), M'') = \emptyset$ (resulting from orienting $Comp(alice) - g'_{1|C'_1} - Rev$ as $Comp(alice) - g'_{1|C'_1} \rightarrow Rev$) and $Pa_2(Comp(alice), M'') = \{Rev\}$ (resulting from orienting $Comp(alice) - g'_{1|C'_1} - Rev$ as $Comp(alice) \leftarrow g'_{1|C'_1} - Rev$). Depending on whether $Comp(alice)$ actually has a parent or not, the post-intervention distribution is different and hence, Alg. 1 returns a set containing two possible results for the query $P(Rev \mid do(Comp(alice) = high))$.

Note that the model shown in Fig. 3 is more compact than the fully grounded model in Fig. 2 as Alg. 1 only grounds necessary parts of the model. Algorithm 1 can further benefit from interventions on multiple indistinguishable objects under the assumption that the graph structure is identical for all groundings. For example, when considering the effect of a training course given to multiple employees on the revenue of the company, even if there are hundreds of employees, after intervening on their competence there are still only two nodes representing their competences in the graph, namely one node for all employees that have received the training course and another node for the remaining employees. In this case, Alg. 1 has to consider only two possible parent sets regardless of the number of employees while there are $2^{|\text{dom}(E)|}$ possible parent sets in an equivalent propositional model to consider (where $|\text{dom}(E)|$ is the number of employees).

Corollary 1. *Let $M = (\mathbf{A} \cup \mathbf{G}, \mathbf{E})$ be a PPCFG. When intervening on a non-grounded PRV $A(L_1, \dots, L_j) \in \mathbf{A}$, under the assumption that the graph structure is identical for all groundings, it holds that*

1. Alg. 1 considers $O(2^{|\text{Ne}(A, M)|})$ possible parent sets in the worst case, and
2. in a propositional model, $O(2^{\prod_{R \in gr(A)} |\text{Ne}(R, gr(M))|})$ possible parent sets have to be considered in the worst case.

Finally, note that Alg. 1 can handle multiple query variables at once instead of a single query variable Q (the query variables are then not summed out in Line 9).

6 Conclusion

We introduce PPCFGs as probabilistic relational models that allow to incorporate partial causal knowledge, thereby enabling lifted causal inference without the need for a fully specified causal model. A lifted representation such as a PPCFG is more expressive than a propositional model and allows for tractable inference with respect to domain sizes of logvars. We further present an algorithm to efficiently compute the effect of joint interventions in PPCFGs (i.e., on a lifted level). Our proposed algorithm is also able to efficiently deal with interventions on PRVs representing sets of indistinguishable randvars.

In future work, we aim to investigate the effect of interventions in a relational model with mutual interdependencies in form of bidirectional edges. We conjecture that randvars with mutual interdependencies can be collapsed into a single node in the graph such that our proposed algorithm can still be applied. Another interesting direction for future work is to allow for hidden confounders.

Acknowledgements

This work is funded by the BMBF project AnoMed 16KISA057. This preprint has not undergone peer review or any post-submission improvements or corrections. The Version of Record of this contribution is published in *Lecture Notes in Computer Science, Volume 15350*, and is available online at https://link.springer.com/chapter/10.1007/978-3-031-76235-2_20.

References

1. Ahmadi, B., Kersting, K., Mladenov, M., Natarajan, S.: Exploiting Symmetries for Scaling Loopy Belief Propagation and Relational Training. *Machine Learning* **92**, 91–132 (2013)
2. Braun, T., Möller, R.: Lifted Junction Tree Algorithm. In: Proceedings of KI 2016: Advances in Artificial Intelligence (KI-16). pp. 30–42. Springer (2016)
3. Braun, T., Möller, R.: Parameterised Queries and Lifted Query Answering. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-18). pp. 4980–4986. IJCAI Organization (2018)
4. De Salvo Braz, R., Amir, E., Roth, D.: Lifted First-Order Probabilistic Inference. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI-05). pp. 1319–1325. Morgan Kaufmann Publishers Inc. (2005)
5. De Salvo Braz, R., Amir, E., Roth, D.: MPE and Partial Inversion in Lifted Probabilistic Variable Elimination. In: Proceedings of the 21th National Conference on Artificial Intelligence (AAAI-06). pp. 1123–1130. AAAI Press (2006)
6. Frey, B.J.: Extending Factor Graphs so as to Unify Directed and Undirected Graphical Models. In: Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI-03). pp. 257–264. Morgan Kaufmann Publishers Inc. (2003)

7. Guo, R., Perkovic, E.: Minimal Enumeration of All Possible Total Effects in a Markov Equivalence Class. In: Proceedings of The 24th International Conference on Artificial Intelligence and Statistics (AISTATS-21). pp. 2395–2403. PMLR (2021)
8. Kersting, K., Ahmadi, B., Natarajan, S.: Counting Belief Propagation. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI-09). pp. 277–284. AUAI Press (2009)
9. Kisiński, J., Poole, D.: Constraint Processing in Lifted Probabilistic Inference. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI-09). pp. 293–302. AUAI Press (2009)
10. Liu, Y., Fang, Z., He, Y., Geng, Z.: Collapsible IDA: Collapsing Parental Sets for Locally Estimating Possible Causal Effects. In: Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI-20). pp. 290–299. PMLR (2020)
11. Luttermann, M., Braun, T., Möller, R., Gehrke, M.: Colour Passing Revisited: Lifted Model Construction with Commutative Factors. In: Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI-24). pp. 20500–20507. AAAI Press (2024)
12. Luttermann, M., Hartwig, M., Braun, T., Möller, R., Gehrke, M.: Lifted Causal Inference in Relational Domains. In: Proceedings of the 3rd Conference on Causal Learning and Reasoning (CLear-24). pp. 827–842. PMLR (2024)
13. Luttermann, M., Machemer, J., Gehrke, M.: Efficient Detection of Commutative Factors in Factor Graphs. In: Proceedings of the Twelfth International Conference on Probabilistic Graphical Models (PGM-2024). pp. 38–56. PMLR (2024)
14. Luttermann, M., Machemer, J., Gehrke, M.: Efficient Detection of Exchangeable Factors in Factor Graphs. In: Proceedings of the 37th International FLAIRS Conference (FLAIRS-24). Florida Online Journals (2024)
15. Luttermann, M., Möller, R., Gehrke, M.: Lifting Factor Graphs with Some Unknown Factors. In: Proceedings of the 17th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-23). pp. 337–347. Springer (2023)
16. Maathuis, M.H., Kalisch, M., Bühlmann, P.: Estimating High-Dimensional Intervention Effects from Observational Data. *The Annals of Statistics* **37**, 3133–3164 (2009)
17. Meert, W., Taghipour, N., Blockeel, H.: First-Order Bayes-Ball. In: Proceedings of the 14th European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD-10). pp. 369–384. Springer (2010)
18. Milch, B., Zettlemoyer, L.S., Kersting, K., Haimes, M., Kaelbling, L.P.: Lifted Probabilistic Inference with Counting Formulas. In: Proceedings of the 23th AAAI Conference on Artificial Intelligence (AAAI-08). pp. 1062–1068. AAAI Press (2008)
19. Nandy, P., Maathuis, M.H., Richardson, T.S.: Estimating the Effect of Joint Interventions from Observational Data in Sparse High-Dimensional Settings. *The Annals of Statistics* **45**, 647–674 (2017)
20. Niepert, M., Van den Broeck, G.: Tractability through Exchangeability: A New Perspective on Efficient Probabilistic Inference. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI-14). pp. 2467–2475. AAAI Press (2014)
21. Pearl, J.: Fusion, Propagation, and Structuring in Belief Networks. *Artificial Intelligence* **29**, 241–288 (1986)
22. Pearl, J.: *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edn. (2009)
23. Pearl, J., Glymour, M., Jewell, N.P.: *Causal Inference in Statistics: A Primer*. Wiley, 1st edn. (2016)

24. Perkovic, E.: Identifying Causal Effects in Maximally Oriented Partially Directed Acyclic Graphs. In: Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI-20). pp. 530–539. PMLR (2020)
25. Poole, D.: First-Order Probabilistic Inference. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03). pp. 985–991. Morgan Kaufmann Publishers Inc. (2003)
26. Shachter, R.D.: Bayes-Ball: Rational Pastime (For Determining Irrelevance and Requisite Information in Belief Networks and Influence Diagrams). In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98). pp. 480–487. Morgan Kaufmann Publishers Inc. (1998)
27. Spirtes, P., Glymour, C., Scheines, R.: Causation, Prediction, and Search. MIT Press, 2nd edn. (2000)
28. Taghipour, N., Fierens, D., Davis, J., Blockeel, H.: Lifted Variable Elimination: Decoupling the Operators from the Constraint Language. *Journal of Artificial Intelligence Research* **47**, 393–439 (2013)
29. Winn, J.: Causality with Gates. In: Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS-12). pp. 1314–1322. PMLR (2012)