Restrictions of PCBNs for integration-free computations

Alexis Derumigny[*], Niels Horsman[†], Dorota Kurowicka[‡]

October 7, 2025

Abstract

The pair-copula Bayesian Networks (PCBN) are graphical models composed of a directed acyclic graph (DAG) that represents (conditional) independence in a joint distribution. The nodes of the DAG are associated with marginal densities, and arcs are assigned with bivariate (conditional) copulas following a prescribed collection of parental orders. The choice of marginal densities and copulas is unconstrained. However, the simulation and inference of a PCBN model may necessitate possibly high-dimensional integration.

We present the full characterization of DAGs that do not require any integration for density evaluation or simulations. Furthermore, we propose an algorithm that can find all possible parental orders that do not lead to (expensive) integration. Finally, we show the asymptotic normality of estimators of PCBN models using stepwise estimating equations. Such estimators can be computed effectively if the PCBN does not require integration. A simulation study shows the good finite-sample properties of our estimators.

Keywords: Pair-copula Bayesian Networks, graphical models, parental orders, stepwise inference.
MSC (2020): 62H22, 62H05 (Primary), 62H12 (Secondary).

1 Introduction

One of the main goals of statistics is to recover the unknown distribution of a random vector $\mathbf{X}$, often represented by its density $f_{\mathbf{X}}$ with respect to some dominating measure. Because of the curse of dimensionality, this is a hard task in general, and one of the way to make this feasible is to use Bayesian Networks (BNs). BNs are composed of a direct acyclic graph (DAG) where the nodes correspond to each of the random variables of $\mathbf{X}$ and the arcs encode the dependence structure of these variables. An extremely attractive feature of these models is their ability to represent complex dependencies in an intuitive way. This is especially important for practitioners, who can easily describe their problems and rely on a solid mathematical theory and many computer implementations of BNs. These models have been applied in a wide variety of fields including medicine, finance, genetics, and forensic science ([16]).

[*]Department of Applied Mathematics, Delft University of Technology, Delft, The Netherlands. E-mail address: a.f.f.derumigny@tudelft.nl

[†]Department of Applied Mathematics, Delft University of Technology, Delft, The Netherlands

[‡]Department of Applied Mathematics, Delft University of Technology, Delft, The Netherlands. E-mail address: d.kurowicka@tudelft.nl

Each component of $\mathbf{X}$ is represented by an element v in a set V, often chosen to be $\{1, \ldots, d\}$ where $\mathbf{X} = (X_1, \ldots, X_d)$. A key property of BNs is that the conditional independencies between components of $\mathbf{X}$, encoded by a graph G with node set V and arc set E, allow for a factorization of the joint density into a product of conditional densities:

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{v \in V} f_{v|pa(v)}(x_v | \mathbf{x}_{pa(v)}),$$

where $f_{v|pa(v)}$ is the conditional density of a node $v \in V$ given its parents $pa(v)$, where a node w is said to be a parent of node v if the arc $w \to v$ is present in G. This factorization allows us to decompose the problem of estimating the (global) high-dimensional density $f_{\mathbf{X}}$ into a set of (local) lower-dimensional problems (a node given its parents). BNs can be used to represent distributions that are purely discrete, purely continuous, or mixed (discrete and continuous with more restrictions, see [15]).

In this paper we consider a particular type of statistical model based on Baysian Networks, which is called pair-copula Bayesian networks (PCBNs). These models were introduced in [12], and were further investigated in [1, 2, 9, 10]. In PCBNs, the conditional densities $f_{v|pa(v)}$ are continuous with respect to Lebesgue's measure and decomposed as a product of bivariate (conditional) copulas. Recall that a copula is a distribution on the unit hypercube with uniform margins, and that, by Sklar's theorem, the joint density $f_{\mathbf{X}}$ can be decomposed as

$$f_{\mathbf{X}}(\mathbf{x}) = c_{\mathbf{X}}\big((F_v(x_v))_{v \in V}\big) \times \prod_{v \in V} f_v(x_v),$$

where $c_{\mathbf{X}}$ is the copula density of $\mathbf{X}$, f_v is the marginal density of X_v and F_v is the marginal cumulative distribution function of X_v. This allows us to separate the estimation of the marginal densities and the copula, which contains all the information about the dependencies between the components of $\mathbf{X}$.

In a PCBN model, each arc $p \to v$ is assigned a continuous (conditional) bivariate copula representing the (conditional) dependence between the random variables X_p and X_v; these copulas must be assigned in a specific manner. If a node v has more than one parent, then a total order $<_v$ is defined over the parental set $pa(v)$. The parents of v are ordered and copulas are then assigned as follows. The arc from the first parent p_1 to v is assigned the copula $C_{p_1,v}$, representing the dependence between X_{p_1} and X_v. Then, the arc from the second parent p_2 to v is assigned the conditional copula $C_{p_2,v|p_1}$, representing the conditional dependence between X_{p_2} and X_v given X_{p_1}. The arc from the third parent p_3 to v is assigned the conditional copula $C_{p_3,v|p_1,p_2}$, representing the conditional dependence between X_{p_3} and X_v given (X_{p_1}, X_{p_2}) and so on. Therefore, the conditional density $f_{v|pa(v)}$ can be decomposed as

$$\begin{aligned} f_{v|pa(v)}(x_v|\mathbf{x}_{pa(v)}) &= f_v(x_v) \cdot c_{p_1,v}(u_{p_1}, u_v) \cdot c_{p_2,v|p_1}(u_{p_2|p_1}, u_{v|p_1} | X_{p_1} = x_{p_1}) \\ &\quad \cdot \ldots \cdot c_{p_m,v|p_1,\ldots,p_{m-1}}(u_{p_m|p_1,\ldots,p_{m-1}}, u_{v|p_1,\ldots,p_{m-1}} \,|\, X_{p_1} = x_{p_1}, \ldots, X_{p_{m-1}} = x_{p_{m-1}}), \end{aligned} \tag{1}$$

where m is the number of parents of v in the graph and $u_{w|S} := F(x_w|\mathbf{X}_S = \mathbf{x}_S)$ for any node $w \in V$ and any set $S \subset V$. Thus, each arc $w \to v$ is assigned the conditional copula $C_{w,v|pa(v \downarrow w)}$, where $pa(v \downarrow w)$ is the set consisting of all parents of v, which are earlier than w according to $<_v$. It has been shown in [12] that such an assignment of copulas is consistent and provides us with a proper joint density $f_{\mathbf{X}}$, whose copula is given by

$$c_{\mathbf{X}}(\mathbf{u}) = \prod_{v\in V} \prod_{w\in pa(v)} c_{w,v|pa(v\downarrow w)}\big(u_{w|pa(v\downarrow w)}\,,\,u_{v|pa(v\downarrow w)}\,\big|\,u_{pa(v\downarrow w)}\big),$$

for $\mathbf{u} \in [0,1]^d$, where d is the dimension of $\mathbf{X}$. Furthermore, if all copulas and marginal distributions in the PCBN are Gaussian, then the PCBN is equivalent to the Gaussian Bayesian Network, see [3, 11, 13, 17].

Parental orders for all nodes are collected in the set $O := (<_v)_{v\in V}$. A PCBN includes the tuple (G, O) where the graph determines (conditional) independencies between elements of the random vector, and the parental orders indicate (conditional) copula assignments. Additionally, the copula types have to be determined and their parameters estimated as well as the marginal distributions of all nodes. PCBNs are much more expensive computationally as compared to Gaussian Bayesian Networks, but they can represent a much more flexible set of dependencies [2].

To compute $c_{\mathbf{X}}(\mathbf{u})$, the terms $u_{w|pa(v\downarrow w)}$ and $u_{v|pa(v\downarrow w)}$ are needed. These conditional margins may require integration.

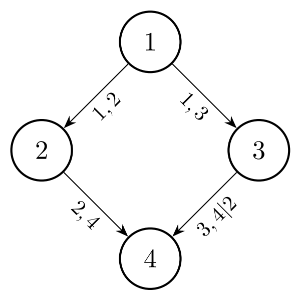


Figure 1: Diamond PCBN on four nodes.

For example, in [1], the graph presented in Figure 1 was found to require integration for any assignment of parental orders O. Note that for this graph we have two possible choices of orders for node 4: $2 <_4 3$ and $3 <_4 2$. When we pick $2 <_4 3$, as presented in Figure 1, the copula density is

$$c(u_1, u_2, u_3, u_4) = c_{1,2}(u_1, u_2) \cdot c_{1,3}(u_1, u_3) \cdot c_{2,4}(u_2, u_4) \cdot c_{3,4|2}(u_{3|2}, u_{4|2}|u_2).$$

The conditional margin $u_{3|2}$, which depends on u_2 and u_3 must in general be computed using integration. This is due to the fact that the marginalization in pair-copula based models cannot be performed analytically. We have that

$$c_{2,3}(u_2, u_3) = \int_0^1 \int_0^1 c(u_1, u_2, u_3, u_4) du_1 du_4 = \int_0^1 c_{1,2}(u_1, u_2) \cdot c_{1,3}(u_1, u_3) du_1$$

which in general cannot be simplified any further.

In this paper the full characterization of graph structures that do not require integration in the density evaluation is presented. We show that if one restricts the structure of G by disallowing certain induced subgraphs then there exists a set of parental orders O for which the evaluation of density is free of integration.

We provide an algorithm that assigns copulas to the arcs of a restricted graph such that the joint density does not require integration.

The rest of the paper is organized as follows. Section 2 presents the background information on DAGs and BNs. Then in Section 3, PCBNs are introduced and the conditions that lead to the need for integration in the density evaluation are studied. Section 4 details the restrictions of PCBNs that are sufficient and necessary for integration-free computations. One of the main theorems of the paper, which guarantees that these restrictions are indeed sufficient for integration-free computations, is proved in Section 5, while supporting lemmas can be found in the appendix. Estimation of PCBN is studied in Section 6. The proposed methodology is implemented in the `R` package `PCBN` [6].

2 Background on DAGs and BNs

This section contains the background information necessary in later parts of the paper.

2.1 Directed graphs

Let $G = (V, E)$ be a directed graph with nodes V and arcs E. We consider only simple graphs without loops and multiple arcs. Moreover, let G^* be the associated undirected graph called **skeleton** of G, obtained from G by removing the directions of the arcs. We say that $G' = (V', E')$ is a **subgraph** of G if $V' \subseteq V$ and $E' \subseteq E$ and for all arcs $w \to v \in E'$ the nodes w and v are in V'. If E' contains all arcs in G between nodes in V', then G' is said to be **induced** by V'. A **path** is a sequence of nodes $(v_1, v_2, \dots, v_n)$ such that $\{v_1, v_2, \dots, v_n\} \subseteq V$ and $\{(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n)\} \subseteq E$ for some integer $n > 0$ called the **length** of the path. A **trail** is a sequence of nodes in G that forms an undirected path in G^*, for which we use the notation $v_1 \rightleftharpoons \cdots \rightleftharpoons v_n$. Moreover, two nodes that are connected by an edge in G^* are called **adjacent**. An arc between non-consecutive nodes in a given trail is referred to as a **chord**. A path of the form $\{v_1, v_2, \dots, v_n, v_1\}$ is called a **cycle**. We call G **acyclic** if it does not contain any cycle. In this case G is a directed acyclic graph (DAG).

For each arc $w \to v \in E$ the node w is said to be the **parent** of v and v is said to be the **child** of w. For a node $v \in V$ the sets containing all its parents and children are denoted by $pa(v)$ and $ch(v)$, respectively. If there exists a path from w to v, then w is said to be an **ancestor** of v and v is said to be a **descendant** of w. For a node $v \in V$ the sets containing all its ancestors and descendants are denoted by $an(v)$ and $de(v)$, respectively.

If a node has at least two parents v_1 and v_2 then we say that (v_1, v, v_2) is a **v-structure** at v and when v has at least two children v_1 and v_2, (v_1, v, v_2) is referred to as a **diverging connection** Moreover, paths (v_1, v_2, v_3) or (v_3, v_2, v_1) in G will be called **serial connections**.

An important concept in graphical models and in particular in BNs whose qualitative part is represented by a directed graph, is that two subsets of nodes can be connected through trails. These trails can be either blocked or activated given another subset of nodes.

Definition 2.1 (d-separation)**.** *Let $G = (V, E)$ be a directed graph and let $S, Y, Z \subseteq V$ be disjoint sets. Then, Z is said to* ***d-separate*** *S and Y in G, denoted by* $\mathrm{dsep}(S, Y|Z)$*, if every trail $v_1 \rightleftharpoons v_2 \rightleftharpoons \cdots \rightleftharpoons v_n$ with $v_1 \in S$ and $v_n \in Y$ contains at least one node v_i satisfying one of the following conditions:*

- *The trail forms a v-structure at v_i, i.e. $v_{i-1} \to v_i \leftarrow v_{i+1}$, and the set $\{v_i\} \sqcup de(v_i)$ is disjoint from Z.*
- *The trail does not contain a v-structure at v_i and $v_i \in Z$.*

If a trail satisfies one of the conditions above, it is said to be ***blocked*** *by Z, else it is* ***activated*** *by Z. Furthermore, if S and Y are not d-separated by Z, we use the notation* $\cancel{\mathrm{dsep}}(S, Y|Z)$*. Moreover, if a set S or Y is empty, then by convention* $\mathrm{dsep}(S, Y|Z)$ *holds.*

2.2 Bayesian networks

A graphical model is a representation of the distribution of a multivariate random vector $\mathbf{X} = (X_1, \ldots, X_d)$ in terms of a graph. Each node $v \in V$ corresponds to a univariate random variable X_v, which is the v-th component of $\mathbf{X}$. In this paper all random vectors are assumed to be absolutely continuous. We denote by f_v the probability density function (pdf) of X_v. For $K \subseteq V$ we write $\mathbf{X}_K := (X_v)_{v \in K}$ and its pdf is denoted by f_K. The cardinal of K is denoted by $|K|$. Furthermore, the pdf of a random variable X_v conditional on $\mathbf{X}_K$ with $K \subseteq V \setminus \{v\}$ is denoted by $f_{v|K}$ and the corresponding conditional cumulative distribution function is denoted by $F_{v|K}$.

Definition 2.2 (Bayesian network)**.** *A* ***Bayesian network*** *(BN) is a graphical model composed of*

- *a DAG $G = (V, E)$ where the nodes correspond to univariate random variables and the arcs describe the conditional independence through d-separation, in the sense that for any disjoints sets $S, Y, Z \subseteq V$,* $\mathrm{dsep}(S, Y|Z)$ *implies that $\mathbf{X}_S$ and $\mathbf{X}_Y$ are independent given $\mathbf{X}_Z$;*
- *a sequence of conditional densities $(f_{v|pa(v)})_{v \in V}$.*

The set of conditional independence statements given by E allows for the decomposition of the joint density of $\mathbf{X}$ as a product of the specified conditional densities:

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{v \in V} f_{v|pa(v)}(x_v|\mathbf{x}_{pa(v)}). \tag{2}$$

Note that different graphical structures can induce the same set of conditional independence statements. Such graphical structures are then called **equivalent**.

As seen in the second point of the definition, a BN require the specification of all conditional densities $f_{v|pa(v)}$. The most popular BNs are discrete and Gaussian BNs, i.e. where each conditional density is either a density with respect to the counting measure (discrete case) or a Gaussian density with respect to Lebesgue's measure (Gaussian case). Nevertheless, in practice it is rare that random variables follow Gaussian distributions, and it is necessary to have more flexible models that can adequately represent real-life distributions. This is why, in this paper, we consider a copula-based Bayesian Network, which is presented next.

3 Pair-Copula Bayesian Networks

3.1 Introduction

Each conditional density $f_{v|pa(v)}$ in the density decomposition (2), can be rewritten as a product of the marginal density f_v and the (conditional) bivariate copula densities as seen in (1): the arc $w \to v$ is assigned the copula $c_{w,v|pa(v\downarrow w)}$, where sets $pa(v \downarrow w)$ and $pa(v \downarrow w)$ defined below.

Definition 3.1 (Parental order)**.** *Let $G = (V, E)$ be a directed graph and $v \in V$ be a node. A **parental order** of v is a total order on the set $pa(v)$ denoted by $<_v$. For all $w \in pa(v)$, the set of parents of v **strictly up to** w (respectively, **after** w) is defined as $pa(v \downarrow w) := \{z \in pa(v);\, z <_v w\}$ (respectively, $pa(v \uparrow w) := \{z \in pa(v);\, w <_v z\}$).*

The formal definition of the PCBN including the parental order for each node is presented.

Definition 3.2 (Pair-copula Bayesian network)**.** *A **pair-copula Bayesian network** is defined as a collection $(G, O, (f_v)_{v\in V}, (C_{w,v|pa(v\downarrow w)})_{w\to v\in E})$ where*

- *the pair (G, O), called the **structure of the PCBN** consists of a DAG G and a collection of orderings $O = (<v)_{v\in V}$,*
- *$(f_v)_{v\in V}$ is a collection of univariate densities,*
- *$(C_{w,v|pa(v\downarrow w)})_{w\to v\in E}$ is a collection of (conditional) copulas.*

The set of conditional independencies allows for the decomposition of the joint density of a PCBN as a product of the marginal densities and copulas;

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{v\in V} f_v(x_v) \prod_{w\in pa(v)} c_{w,v|pa(v\downarrow w)}\big(F_{w|pa(v\downarrow w)}(x_w|\mathbf{x}_{pa(v\downarrow w)})\,,\, F_{v|pa(v\downarrow w)}(x_v|\mathbf{x}_{pa(v\downarrow w)})\,\big|\,\mathbf{x}_{pa(v\downarrow w)}\big). \quad (3)$$

To simplify the notation, for any node $v \in V$ and for any set $S \subset V$, we define $u_{v|S} := \mathbb{P}(U_v \le u_v|\mathbf{U}_S = \mathbf{u}_s)$. In particular, we will denote $F_{v|pa(v\downarrow w)}(x_w|\mathbf{x}_{pa(v\downarrow w)})$ by $u_{v|pa(v\downarrow w)}$.

The graph in Figure 2, with order $1 <_4 2 <_4 < 3$, has copulas $c_{1,4}$, $c_{2,4|1}$ and $c_{3,4|1,2}$ assigned to its arcs. The corresponding copula density is as follows:

$$c_{1,2,3,4}(u_1, u_2, u_3, u_4) = c_{1,4}(u_1, u_4) \cdot c_{2,4|1}(u_2, u_{4|1}|u_1) \cdot c_{3,4|1,2}(u_3, u_{4|1,2}|u_1, u_2).$$

We see that X_1, X_2 and X_3 are mutually independent.

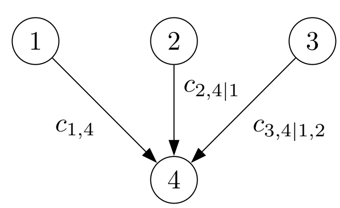


Figure 2: PCBN where node X_4 has corresponding parental order $1 <_4 2 <_4 3$.

Note that we do not assume that the copula $C_{w,v|pa(v\downarrow w)}$ is constant with respect to $\mathbf{x}_{pa(v\downarrow w)}$. Such assumption is known as the "simplifying assumption" (see e.g. [4] for a review), and is not needed in this paper. All the results presented in this paper are valid in both cases whether the simplifying assumption is made or not.

From Equation (3), it is clear that the density can be written as a product of all marginal densities of $\mathbf{X}$, and a copula density. This copula density $c_{\mathbf{X}}$ corresponding to $f_{\mathbf{X}}$ is then decomposed as a product of (conditional) copula densities assigned to arcs in the graph, by

$$c_{\mathbf{X}}(\mathbf{u}) = \prod_{v\in V} \prod_{w\in pa(v)} c_{w,v|pa(v\downarrow w)}\big(u_{w|pa(v\downarrow w)}, u_{v|pa(v\downarrow w)} \,|\, \mathbf{u}_{pa(v\downarrow w)}\big). \tag{4}$$

The PCBN structure (G, O) provides us with a collection of conditional copulas which are assigned to arcs in the graph: $(C_{w,v|pa(v\downarrow w)})_{w\to v\in E}$. These copulas are said to be specified by the PCBN. Furthermore, the graph of a PCBN induces conditional independencies between random variables that can be read from the graph through the d-separation. If two nodes are d-separated given set of nodes, then the conditional copula of variables corresponding to these nodes is also known; it is the independence copula. Finally, adding conditionally independent variables to the conditioning set of an already specified copulas yields a copula that is still specified (see Figure 3). Hence, we formalize when $C_{w,v|K}$ is specified.

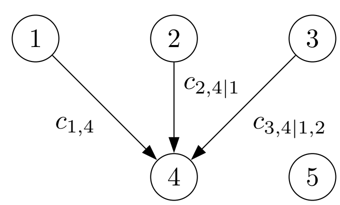


Figure 3: PCBN where $C_{2,4|1,5} = C_{2,4|1}$ is specified.

Definition 3.3. *Consider a PCBN with node set V, and let $w, v \in V$. We say that the following (conditional) copulas are* **specified***:*

(i) $C_{v,w|pa(v\downarrow w)}$ *and* $C_{w,v|pa(v\downarrow w)}$, *if* $w \to v$, *since these copulas are explicitly specified in the PCBN.*

(ii) $C_{w,v|K}$, *where* $K \subseteq V \setminus \{w, v\}$ *and* $\mathrm{dsep}(w, v|K)$. *In this case,* $C_{w,v|K}$ *is known from the graph structure to be the independence copula.*

(iii) $C_{w,v|pa(v\downarrow w)\sqcup J}$, *where* $w \to v$ *and* $J \neq \emptyset$ *is such that* U_v *is independent of* $\mathbf{U}_J$ *given* $(U_w, \mathbf{U}_{pa(v\downarrow w)})$, *i.e.* $\mathrm{dsep}\big(v, J \,\big|\, pa(v \downarrow w) \sqcup \{w\}\big)$. *Then* $C_{w,v|pa(v\downarrow w)\sqcup J}$ *is equal to the explicitly specified copula* $C_{v,w|pa(v\downarrow w)}$.

Remark 3.1. *Note that a necessary condition for* $C_{w,v|K}$ *to be specified is that* v *and* w *are either adjacent or d-separated by* K. *Therefore, copulas of the form* $C_{w,v|K}$ *where* v *and* w *are neither adjacent nor d-separated by* K *always require integration.*

Remark 3.2. *Note that copulas* $C_{w,v|K}$ *that are obtained from* $C_{w,v|pa(v\downarrow w)}$ *by removing nodes from the conditioning set* $pa(v \downarrow w)$ *are never specified. They always must be computed by integration with respect to* *the nodes that need to be removed. More generally,*

$$w \to v \text{ and } \exists o \in pa(v \downarrow w) \text{ such that } o \notin K \tag{5}$$

is a sufficient condition for $C_{w,v|K}$ to be not specified.

3.2 Problematic conditional margins for PCBNs

Equation (4) requires the computation of the conditional margins $u_{w|pa(v\downarrow w)}$ and $u_{v|pa(v\downarrow w)}$, for $v \in V$ and $w \in pa(v)$. This means that $u_{w|pa(v\downarrow w)}$ and $u_{v|pa(v\downarrow w)}$ must be computed in order to evaluate the density $f_{\mathbf{X}}$. In this section, we will see two examples where the term $u_{w|pa(v\downarrow w)}$ cannot be computed without using integration.

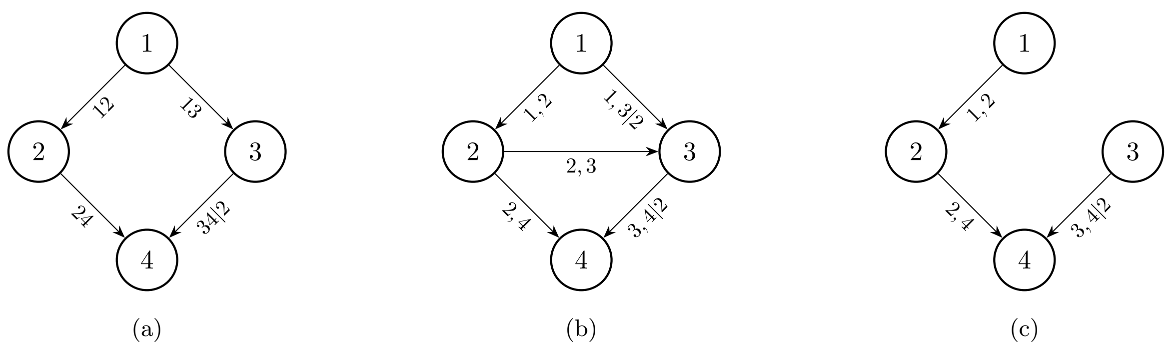


Figure 4: PCBNs where the computation of $u_{3|2}$ can be done without integrating for (b) and (c), but not for (a).

Example 3.1. *The copula density corresponding to the PCBN in Figure 4a* [1] *is as follows:*

$$c(\mathbf{u}) = c_{1,2}\big(u_1, u_2\big) \cdot c_{1,3}\big(u_1, u_3\big) \cdot c_{2,4}\big(u_2, u_4\big) \cdot c_{3,4|2}\big(u_{3|2}, u_{4|2} \,|\, u_2\big).$$

To compute this copula density, we need the copulas $c_{1,2}, c_{1,3}, c_{2,4}, c_{3,4|2}$, which are already specified, but we also need to compute the conditional marginals $u_{3|2}$ and $u_{4|2}$. By definition, $u_{3|2} = \frac{\partial C_{2,3}(u_2,u_3)}{\partial u_2}$. Since the copula $c_{2,3}$ is not specified by the PCBN, it needs to be computed. We know that

$$c_{2,3}(u_2, u_3) = \int_0^1 c_{1,2,3}(w_1, u_2, u_3) dw_1 = \int_0^1 c_{1,2}\big(w_1, u_2\big) \cdot c_{1,3}\big(w_1, u_3\big)\, \mathrm{d}w_1.$$

As a consequence, we have

$$\begin{aligned} u_{3|2} &= \frac{\partial}{\partial u_2} \int_0^{u_2} \int_0^{u_3} c_{2,3}(w_2, w_3)\, \mathrm{d}w_2\, \mathrm{d}w_3 = \int_0^{u_3} c_{2,3}(u_2, w_3)\, \mathrm{d}w_3 \\ &= \int_0^1 c_{1,2}\big(w_1, u_2\big) \int_0^{u_3} c_{1,3}\big(w_1, w_3\big) dw_3\, \mathrm{d}w_1 \\ &= \int_0^1 c_{1,2}(w_1, u_2) \cdot \frac{\partial C_{1,3}(w_1, u_3)}{\partial u_1}\, \mathrm{d}w_1 = \int_0^1 c_{1,2}(w_1, u_2) \cdot h_{1,3}(w_1, u_3)\, \mathrm{d}w_1. \end{aligned}$$

[1] This example was already shortly discussed in the Introduction.

Hence, integration is needed, since $c_{1,2}$ and $C_{1,3}$ can be specified arbitrarily. Note that the conditional margin $u_{4|2}$ does not pose a problem since it can be computed using the copula $C_{2,4}$, which is specified by the PCBN.

On the contrary, for the PCBNs in Figures 4b and 4c there is no problem with computing the conditional margin $u_{3|2}$. Indeed, for the PCBN in Figure 4b the copula $c_{2,3}$ is assigned to the arc $v_2 \to v_3$. For the PCBN in Figure 4c, $c_{2,3}$ is also specified because it is the independent copula, due to $\operatorname{dsep}(2, 3 \,|\, \emptyset)$.

It is important to notice that even for the same graph structure, certain ordering O necessitate integration, while other does not. For example, if the PCBN Figure 4b, we instead of $2 <_3 1$ choose the order $1 <_3 2$, then the copula $c_{2,3}$ is not specified by the PCBN. Indeed, in this case copulas $c_{1,3}$ and $c_{2,3|1}$ are specified. Hence, the computation of conditional margin $u_{3|2}$ requires integration.

Example 3.2. *Consider the PCBN in Figure 5. The orderings at nodes v_4 and v_5 have been chosen and we need to determine the order at node v_3. The v-structures at nodes v_4 and v_5 require us to compute the conditional margins $u_{3|1}$ and $u_{2|3}$, respectively. The former implies that $c_{1,3}$ must be specified, and thus we must have $1 <_3 2$. But, the latter requires $c_{2,3}$ to be specified, implying that $2 <_3 1$. Since we cannot have both, there does not exist a suitable ordering for node v_3.*

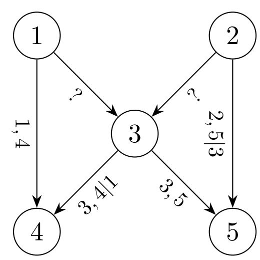


Figure 5: PCBN with interfering v-structures.

If G contains three v-structures that interact in a similar fashion as in Example 3.2, then the joint density will require integration for any choice of O. Such problematic set of three v-structures will be called interfering v-structures (see Definition 3.7 below).

All copulas $C_{w_i,v|w_1,\dots,w_{i-1}}$, where $pa(v) = \{w_1, ..., w_s\}$ are specified by construction. Hence computation of $u_{v|pa(v\downarrow w)}$ is in general not problematic. The main obstacle is in the computation of terms of the form $u_{w|pa(v\downarrow w)}$. These conditional margins are computed recursively. Let $v \in V$ and $K \subseteq V \setminus \{v\}$ then

$$u_{v|K} \quad = \quad \frac{\partial C_{k,v|K\setminus k}(u_{k|K\setminus k}, u_{v|K\setminus k} \,|\, \mathbf{u}_{K\setminus k})}{\partial u_{k|K\setminus k}}. \tag{6}$$

First we formally define such recursions, which will be used repeatedly throughout the paper.

Definition 3.4. *Consider a PCBN with a node set V. Let $v \in V$ and $K \subseteq V \setminus \{v\}$. $\mathcal{R}(v|K)$ is **the set of recursions** for the computation of $u_{v|K}$ in (6) if*

(i) $\mathcal{R}(v|\emptyset) := \{u_v\}$.

(ii) $\mathcal{R}(v|K) := \bigsqcup_{k\in K} \big\{\big(u_{v|K}, k, C_{v,k|K\setminus k}, R_v, R_k\big) : R_v \in \mathcal{R}(v|K \setminus k), R_k \in \mathcal{R}(k|K \setminus k)\big\}$.

Remark 3.3. *From Definition 3.4 it follows that any recursion $R \in \mathcal{R}(v|K)$ requires the choice of some $k \in K$, which determines a conditional copula $C_{v,k|K\setminus k}$, and a recursion $R_v \in \mathcal{R}(v|K \setminus k)$. From this branch of the recursion, there exists an ordering $(k_1, \dots, k_{|K|})$ of K such that copulas*

$$\forall i \in \{1, \dots, |K|\},\ C_{v,k_i|K\setminus\{k_i,\dots,k_{|K|}\}} = C_{v,k_i|k_1,\dots,k_{i-1}} \text{ appear in } R.$$

The ordering of K depends on the choice of the recursion $R \in \mathcal{R}(v|K)$.

Definition 3.5. *We say that a recursion $R \in \mathcal{R}(v|K)$ is* ***proper*** *if all (conditional) copulas appearing in R are specified. Moreover, $u_{v|K}$* **does not require integration** *if there exist a proper recursion $R \in \mathcal{R}(v|K)$.*

Remark 3.4. *From Definition 3.5 it follows that $u_{v|K}$ does not require integration if there exists $k \in K$ such that*

1. *$C_{k,v|K\setminus k}$ is specified by the PCBN;*
2. *$u_{k|K\setminus k}$ and $u_{v|K\setminus k}$ do not require integration.*

Remark 3.5. *If a recursion to compute $u_{v|pa(v\downarrow w)}$ is proper, then the ordering of $pa(v \downarrow w)$ presented in Remark 3.3 must be the one induced by $<_V$.*

Computation of $u_{v|pa(v\downarrow w)}$ does not require integration. We show next that computation of the joint density is easy when $u_{w|pa(v\downarrow w)}$ can be computed without integration.

Lemma 3.1. *The joint copula density $c_{\mathbf{X}}(\mathbf{u})$ (and therefore $f_{\mathbf{X}}(\mathbf{x})$) of a PCBN can be computed without integration if and only if*

$$\forall v \in V, \forall w \in pa(v), \text{ the conditional margin } u_{w|pa(v\downarrow w)} \text{ does not require integration.} \tag{7}$$

Proof. Assume that Condition (7) is not satisfied. Then one of the terms $u_{w|pa(v\downarrow w)}$ in Equation (4) must be computed with integration, and therefore $c_{\mathbf{X}}(\mathbf{u})$ needs integration. Assume now that Condition (7) is satisfied. Since all pair-copula appearing in (4) are specified, and all conditional margins $u_{w|pa(v\downarrow w)}$ can be computed without integration by Condition (7), there only remains to prove that all terms of the form $u_{v|pa(v\downarrow w)}$ can be computed without integration.

Note that only nodes v such that $|pa(v)| > 0$ appear in the factorization (4). Let $v \in V$ be such a node and let $pa(v) := (w_1, \dots, w_n)$ be ordered according to $<_V$. We show by induction on $i \in \{1, \dots, n\}$ that the conditional margins $u_{v|pa(v\downarrow w_i)}$ do not require integration. For $i = 1$, the statement holds since $u_{v|pa(v\downarrow w_1)} = u_v$. Now, suppose that $u_{v|pa(v\downarrow w_{i-1})}$ can be computed without integration. The conditional margin $u_{v|pa(v\downarrow w_i)}$ is

$$u_{v|pa(v\downarrow w_i)} = \frac{\partial C_{w_{i-1},v|pa(v\downarrow w_{i-1})}}{\partial u_{w_{i-1}|pa(v\downarrow w_{i-1})}}\big(u_{w_{i-1}|pa(v\downarrow w_{i-1})}, u_{v|pa(v\downarrow w_{i-1})} \,|\, \mathbf{u}_{pa(v\downarrow w_{i-1})}\big)$$

where the copula $C_{w_{i-1},v|pa(v\downarrow w_{i-1})}$ is specified by the PCBN. The conditional margin $u_{v|pa(v\downarrow w_{i-1})}$ does not require integration by the induction hypothesis. Moreover, $u_{w_{i-1}|pa(v\downarrow w_{i-1})}$ does not require integration by Condition (7). Hence, the conditional margin $u_{v|pa(v\downarrow w_i)}$ does not require integration. □

We now establish a convenient lemma, which gives a sufficient condition for (7) not to be satisfied, i.e. for $u_{w|pa(v\downarrow w)}$ to require integration. The condition require existence of a node z for which none of the conditional copulas $C_{v,z|K}$ are specified. Indeed, in such case none of the recursions can be proper, as a copula of this form necessarily appears at some point at these recursions.

Lemma 3.2. *Consider a PCBN with node set V, and let $v \in V$ and $w \in pa(v)$. The computation of the conditional margin $u_{w|pa(v\downarrow w)}$ requires integration if*

$$\exists z \in pa(v \downarrow w),\ \forall K \subseteq pa(v \downarrow w) \setminus \{z\},\ C_{z,w|K} \text{ is not specified.} \tag{8}$$

Proof. According to Remark 3.3, we know that a copula of the form $C_{z,w|K}$ for some $K \subseteq pa(v \downarrow w) \setminus \{z\}$ necessarily appears at some point in any recursion $R \in \mathcal{R}(w|pa(v \downarrow w))$ to compute $u_{w|pa(v\downarrow w)}$. So, $u_{w|pa(v\downarrow w)}$ requires integration. $\square$

3.3 Active cycles

The PCBN in Figure 4a is an example of a more general structure which we call active cycle. In this diamond-type graph, the v-structure $v_2 \to v_4 \leftarrow v_3$ is combined with a diverging connection at node v_1. Together they form the cycle $v_4 - v_2 - v_1 - v_3 - v_4$ in the corresponding undirected graph. Such undirected cycles always lead to a problematic conditional margin. The diverging connection at v_1 can be replaced by a serial connection and the problem with computing the conditional margin remains. The general definition of this problematic structure is given below.

Definition 3.6. *Let G be a DAG. Consider a node $v \in V$ with distinct parents $w, z \in pa(v)$ which are connected by a trail $w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z$, $n \geq 1$, satisfying the following conditions:*

(i) $w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z$ consists of only diverging or serial connections.

(ii) $v \leftarrow w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z \to v$ contains no chords.

Then the trail $v \leftarrow w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z \to v$ is called an ***active cycle*** *in G. Furthermore, G is said to contain an active cycle.*

The presence of an active cycle in the graph necessitates integration. This statement is proven in Theorem 3.1.

Theorem 3.1. *Let (G, O) be a PCBN. If G contains an active cycle, then the computation of the joint density requires integration.*

Proof. Consider an active cycle in G of the form

$$v \leftarrow w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z \to v.$$

Since w and z are both parents of v, we have either $w <_v z$ or $z <_v w$. Let us assume that $z <_v w$. We want to prove that the margin $u_{w|pa(v\downarrow w)}$ requires integration. Due to Lemma 3.2, it is sufficient to show that for any $K \subseteq pa(v \downarrow w) \setminus \{z\}$ the copula $C_{z,w|K}$ is not specified by the PCBN. Consider an arbitrary $K \subseteq pa(v \downarrow w) \setminus \{z\}$.

Note that w and z are not adjacent and the trail between w and z is not blocked by any subset of nodes in $pa(v)$, due to the existence of trail $w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons z$ without a cord. Since $K \subseteq pa(v \downarrow w) \subseteq pa(v)$, we have that $\cancel{\text{dsep}}(w, z \mid K)$. Thus, by Remark 3.1, the copula $C_{z,w|K}$ is not specified. □

3.4 Interfering v-structures

In Figure 5, we showed another example of PCBN that requires integration for any choice of the parental ordering. This structure is formally defined below.

Definition 3.7. *Consider a PCBN with node set V. Assume that there exist nodes $v_1, v_2, v_3, v_4, v_5 \in V$, satisfying the following conditions:*

- $v_3 \to v_4$, $v_3 \to v_5$
- $v_1 \to v_3$, $v_1 \to v_4$, *and* $v_1 \not\to v_5$
- $v_2 \to v_3$, $v_2 \to v_5$ *and* $v_2 \not\to v_4$.

*Then, the nodes v_1, v_2, v_3, v_4 and v_5 are said to be **interfering v-structures**. Moreover, G is said to contain interfering v-structures.*

Note that one or both of the arcs $v_1 \to v_2$ or $v_2 \to v_1$ can be added to the DAG in Figure 5 and the interfering v-structure will remain. However, this will not be the case if at least one of the arc $v_1 \to v_5$ or $v_2 \to v_4$ is added. Removal of any of the arcs present in DAG in Figure 5 alleviates the problem of the need to integrate.

We will prove next that for any graph containing interfering v-structures, the computation of the joint density requires integration for any choice of O.

Theorem 3.2. *Consider a PCBN with DAG G. If G contains interfering v-structures, then the computation of the joint density requires integration.*

Proof. Let $v_1, v_2, v_3, v_4, v_5 \in V$ be nodes that form one of the interfering v-structures in G. We have eight distinct cases concerning constraints on parental orderings of nodes v_3, v_4, and v_5, these are:

$$
\begin{aligned}
&v_1 <_{v_3} v_2,\ v_1 <_{v_4} v_3 \text{ and } v_2 <_{v_5} v_3, && v_2 <_{v_3} v_1,\ v_1 <_{v_4} v_3 \text{ and } v_2 <_{v_5} v_3,\\
&v_1 <_{v_3} v_2,\ v_1 <_{v_4} v_3 \text{ and } v_3 <_{v_5} v_2, && v_2 <_{v_3} v_1,\ v_1 <_{v_4} v_3 \text{ and } v_3 <_{v_5} v_2,\\
&v_1 <_{v_3} v_2,\ v_3 <_{v_4} v_1 \text{ and } v_2 <_{v_5} v_3, && v_2 <_{v_3} v_1,\ v_3 <_{v_4} v_1 \text{ and } v_2 <_{v_5} v_3,\\
&v_1 <_{v_3} v_2,\ v_3 <_{v_4} v_1 \text{ and } v_3 <_{v_5} v_2, && v_2 <_{v_3} v_1,\ v_3 <_{v_4} v_1 \text{ and } v_3 <_{v_5} v_2.
\end{aligned}
$$

Since all cases are analogous, we only consider the case when: $v_1 <_{v_3} v_2$, $v_1 <_{v_4} v_3$ and $v_2 <_{v_5} v_3$. Then we have

$$
\begin{aligned}
&v_1 \in pa(v_4 \downarrow v_3) \text{ and } v_1 \notin pa(v_5 \downarrow v_3) \subseteq pa(v_5),\\
&v_2 \in pa(v_5 \downarrow v_3) \text{ and } v_2 \notin pa(v_4 \downarrow v_3) \subseteq pa(v_4).
\end{aligned}
$$

We apply Lemma 3.2, with $z = v_2$, $w = v_3$ and $v = v_5$ to find that the computation of $u_{v_3|pa(v_5\downarrow v_3)}$ requires integration.

Let $K \subseteq pa(v_5 \downarrow v_3) \setminus \{v_2\}$. Since there is the arc $v_2 \to v_3$, the nodes v_2 and v_3 are not d-separated given any subset of $V \setminus \{v_2, v_3\}$. Thus, the copula $C_{v_2,v_3|K}$ is not the independence copula.

The arc $v_2 \to v_3$ has the assigned copula $C_{v_2,v_3|pa(v_3\downarrow v_2)}$. Note that $v_1 \in pa(v_3 \downarrow v_2)$, but $v_1 \notin pa(v_5)$. Therefore, $v_1 \notin K$ as $K \subseteq pa(v_5)$. Hence, the copula $C_{v_2,v_3|K}$ is also not specified by an arc. □

4 Restricted PCBNs

In the previous Sections 3.3 and 3.4, we have shown that PCBNs containing active cycles and/or interfering v-structures necessitate integration. We now announce the main result, which is that these are the only graphical structures for which integration is needed.

Theorem 4.1. *Let G be a DAG. There exists a collection of orderings O such that the computation of the joint density of the PCBN (G, O) does not require integration if and only if G contains no active cycles nor interfering v-structures.*

Proof. The sufficiency is proven using contraposition and applying Theorems 3.1 and 3.2. The necessity is proven by combining Theorems 4.2 and 4.3. □

To prove the necessity in Theorem 4.1, we will demonstrate that for any graph G that does not contain active cycles or interfering v-structures, we can find a collection of orderings O such that in computation of conditional margins integration is not needed. Therefore, we construct an algorithm which is able to find a suitable O for any restricted DAG G.

4.1 Possible candidates and algorithm

The algorithm follows an arbitrary well ordering $(v_1, \ldots, v_n)$ of nodes in V. For any node $v \in V$, a suitable ordering $<_v$ is chosen sequentially. This means that when we arrive at a node v we will have already chosen the order $<_z$ for all nodes in $z \in pa(v)$. The process of finding a suitable order $<_v$ involves growing an ordered set $O_v^k = (o_1, \ldots, o_k)$, referred to as a partial order. This should be interpreted as $o_1 <_v \cdots <_v o_k$.

Definition 4.1 (Partial order)**.** *For a node $v \in V$, an ordered subset of k parents of v will be referred to as a* ***partial order*** *denoted by O_v^k. Thus, we have*

$$O_v^k = (o_1, \ldots, o_k)$$

with $k \leq |pa(v)|$ and $o_i \in pa(v)$ for all $i \in \{1, \ldots, k\}$.

An initial state is $O_v^0 = \emptyset$ to which at each iteration a node from the set $pa(v)$ is added until we have found $O_v^{|pa(v)|}$. A node can be added to a partial order O_v^k if it satisfies certain constraints. Specifically, we can add a $w \in pa(v)$ such that we can compute the conditional margin $u_{w|O_v^k}$ without integration. This motivates the definition.

Definition 4.2. *The set of **possible candidates** for a partial order O_v^k is defined by*

$$PossCand(O_v^k) := \big\{w \in pa(v) \text{ such that } u_{w|O_v^k} \text{ can be computed without integration}\big\}.$$

Therefore, we propose the following Algorithm 1.

Algorithm 1 Finding a suitable O.

Require: restricted DAG G

Ensure: set of orderings O for which we will not require integration

1: **for each** node v in V according to a well-ordering **do**
2:     $O_v^0 \leftarrow \emptyset$
3:     **for each** $k = 0, \ldots, |pa(v)| - 1$ **do**
4:         Compute $PossCand(O_v^k)$.
5:         Choose one element $w \in Poss.Cand(O_v^k)$
6:         Define $O_v^{k+1} := (O_v^k, w)$.
7:     **end for**
8:     Set $<_v$ according to $O_v^{|pa(v)|}$
9: **end for**
10: **return** $O := \{<_v;\, v \in V\}$

Algorithm 1 allows finding all sets of orders O that do not result in integration. This algorithm seems very simple. The main issue here is how to find the set of possible candidates and whether it is non-empty. This will be discussed next.

Theorem 4.2. *Let G be a DAG containing no active cycles nor interfering v-structures. The joint density of any PCBN with DAG G does not requires integration if and only if its set of orders O is one of the possible outputs of Algorithm 1.*

Proof. This is a direct consequence of Lemma 3.1. Indeed, for any node v and any $w \in pa(v)$, there are $k = |pa(v \downarrow w)|$ elements smaller than w (with respect to $<_v$) in $pa(v)$. Remark that $pa(v \downarrow w) = O_v^k$. By definition, $u_{w|pa(v\downarrow w)} = u_{w|O_v^k}$ does not require integration if and only if $w \in PossCand(O_v^k)$, i.e. if and only if it is chosen by the algorithm. □

Before characterizing the set of possible candidates, we announce one important result. Its proof is difficult and is delayed to Section 5.

Theorem 4.3. *Let G be a DAG containing no active cycles nor interfering v-structures. The set $PossCand(O_v^k)$ computed in line 4 of Algorithm 1 is never empty.*

This result implies that we will never encounter a case where there is no possible candidate to be added. Hence, the algorithm will never terminate prematurely nor get "stuck" and will always return a suitable set of orders O.

4.2 B-sets

The construction of B-sets is motivated by observation made in Section 3.4 that the children of a node v with a v-structure can constrain the order of parents of v. This happens because v and its children can have common parents. In the PCBN represented in Figure 4b, we have $pa(v_3) = \{v_1, v_2\}$ and $pa(v_4) = \{v_2, v_3\}$. Hence v_2 is their common parent and it has to be put first in the parental order of v_3. Similarly, for the PCBN in Figure 5 the common parent of v_3 and v_4 is v_1, hence it should be put as first in the parental order of v_3. However, at the same time the common parent of v_3 and v_5 is v_2, which we would need to put first in the parental order of v_3. To formalize these observations we introduce the concept of B-sets.

Definition 4.3 (B-set)**.** *Let $G = (V, E)$ be a DAG. For $v, w \in V$ such that $v \to w$, we denote by*

$$B(v, w) := pa(v) \cap pa(w)$$

*We say that $B(v, w)$ is a **B-set** of v.*

The B-sets will provide us with clear restrictions an order must abide to, so that the joint density does not require integration. It should be noted that a node has as many B-sets as it has children. Some of them can be empty and not all of them have to be distinct. In the lemma below we prove that the B-sets of a node in DAG G are ordered by inclusion if and only if G does not contain interfering v-structures.

Lemma 4.1. *Let $G = (V, E)$ be a DAG. The following two statements are equivalent:*

(i) G does not contain interfering v-structures.

(ii) For all $v_1 \in V$ and $v_2, v_3 \in ch(v_1)$ we have $B(v_1, v_2) \subseteq B(v_1, v_3)$ or $B(v_1, v_3) \subseteq B(v_1, v_2)$.

Proof. If G contains interfering v-structures, then (ii) is violated. For example, in Figure 5 we have $\{v_1\} = B(v_3, v_4) \not\subseteq B(v_3, v_5) = \{v_2\}$ and $B(v_3, v_5) \not\subseteq B(v_3, v_4)$.

If (ii) is violated, then we can find $v_4 \in B(v_1, v_2) \setminus B(v_1, v_3)$ and $v_5 \in B(v_1, v_3) \setminus B(v_1, v_2)$. In this case the nodes $v_1, \ldots, v_5$ are exactly interfering v-structures. ☐

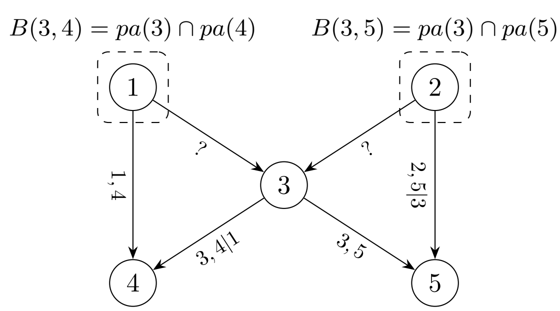


Figure 6: PCBN with interfering v-structures: the B-sets of 3 are not ordered by $\subseteq$.

Thus, if graph G does not contain interfering v-structures, we can order the B-sets corresponding to each node v according to the inclusion order $\subseteq$. The sorted sequence of these subsets with respect to the inclusion relation is referred to as the **B-sets** of v and it determines a partial order of the parental set of v.

Definition 4.4 (B-sets)**.** *Consider a DAG with no interfering v-structures and let $v \in V$ with*

$$Q = Q(v) := \big|\{B(v,w);\, w \in ch(v)\}\big|$$

*the number of distinct B-sets corresponding to v. We denote by $B_1(v), \ldots, B_Q(v)$ the sorted sequence of $\{B(v, v_2);\, v_2 \in ch(v)\}$ in increasing order with respect to $\subseteq$. We also define $B_0(v) := \emptyset$ and $B_{Q+1} := pa(v)$. The sequence $\big(B_q(v)\big)_{q=0,\ldots,Q+1}$ is referred to as the **B-sets** of v.*

Furthermore, for each B-set B_q with $q < Q(v)+1$, we denote by b_q an arbitrary node such that $B_q = B(v, b_q)$. This node b_q may not be unique.

The B-sets introduce the restriction that all nodes in B_q must be smaller than nodes in $B_{q+1} \setminus B_q$ with respect to the order $<_v$. We denote this by $B_q <_v B_{q+1} \setminus B_q$. Hence, we must have

$$B_1 <_v B_2 \setminus B_1 <_v B_3 \setminus B_2 <_v \cdots\cdots\cdots <_v B_{Q+1} \setminus B_Q.$$

We now state the following definition.

Definition 4.5 (Abiding by the B-sets)**.** *Let (G, O) be a PCBN where G contains no interfering v-structures. A parental order $<_v$ is said to **abide** by the B-sets if*

$$B_1 <_v B_2 \setminus B_1 <_v B_3 \setminus B_2 <_v \cdots\cdots\cdots <_v B_{Q+1} \setminus B_Q.$$

Similarly, a set of orders $O := \{<_v;\, v \in V\}$ abides by the B-sets if all its parental orders $<_v$ abide by the B-sets.

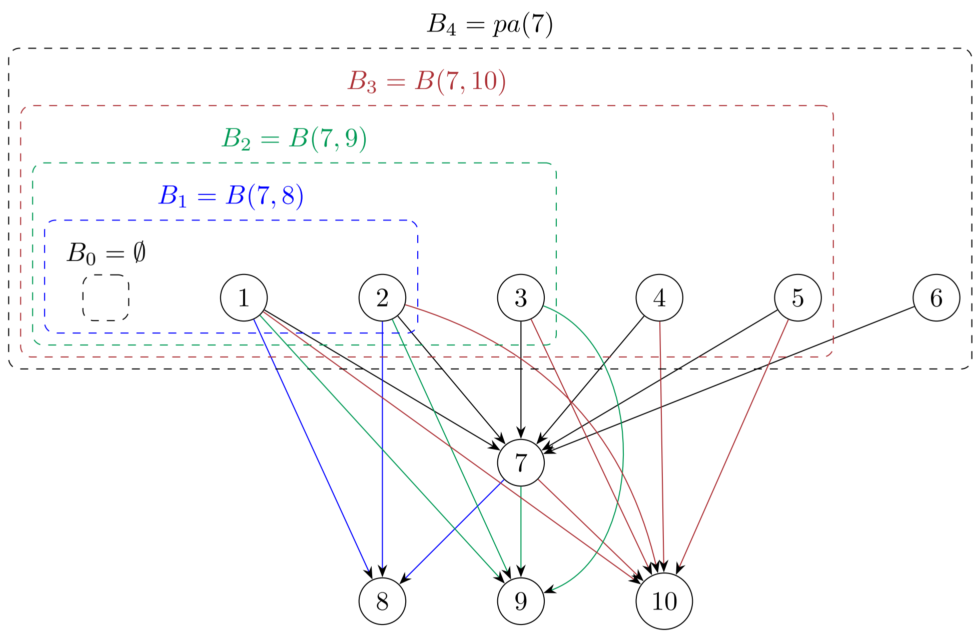
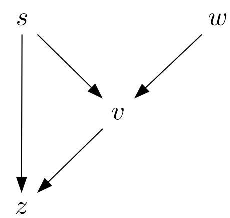


Figure 7: PCBN with ordered B-sets for the node $v = 7$. Colors represent the different B-sets. Any order $<_7$ that abides by the B-sets must start with 1, 2 in any order; then 3; then 4, 5 in any order, and finally 6. This means that the only possible orders $<_7$ that abide by the B-sets are $(1,2,3,4,5,6)$, $(1,2,3,5,4,6)$, $(2,1,3,4,5,6)$ and $(2,1,3,5,4,6)$.

Any PCBN whose set of orders does not abide by the B-sets will require integration.

Lemma 4.2. *Let (G, O) be a PCBN where G contains no active cycles nor interfering v-structures. If O does not abide by the B-sets, then the computation of the joint density requires integration.*

Proof. By assumption, there is a node v in V such that $<_v$ does not abide by the B-sets. Hence, there exist $s, w \in pa(v)$ and a node $z \in ch(v)$, $z \in ch(s)$ and $w \notin pa(z)$ with $w <_v s$. This means that G contains the subgraph below.



The factorization of the joint density requires the computation of the conditional margins $u_{v|pa(z\downarrow v)}$ and $u_{s|pa(z\downarrow s)}$. Both $s <_z v$ or $v <_z s$ are possible but since they are analogous, we only consider the case when $s <_z v$, which implies that $s \in pa(z \downarrow v)$. To compute the joint density, we need the conditional margin $u_{v|pa(z\downarrow v)}$, and we show that this margin requires integration.

To achieve this Lemma 3.2 is applied. Let us consider an arbitrary $K \subseteq pa(z \downarrow v) \setminus \{s\}$. It is sufficient to show that the copula $C_{s,v|K}$ is not specified by the PCBN. Note that since $w <_v s$ then $w \in pa(v \downarrow s)$. However, we have $w \notin pa(z \downarrow v)$, which means $w \notin K$. Therefore, by Remark 3.2, the copula $C_{s,v|K}$ is not specified. □

By Lemma 4.2 a parental order $<_v$ must abide to the B-sets in the sense of Definition 4.5 to prevent integration. Therefore, we obtain the following corollaries.

Corollary 4.1. *All parental orders determined by the algorithm abide by the B-sets.*

Corollary 4.2. *Let x, o, w, q be nodes such that $x <_o w$ and $w \in B(o, q)$, where the order is determined by the algorithm. Then we have $x \in B(o, q)$ and in particular $x \to q$.*

Proof. By Corollary 4.1 all previously determined parental orders chosen by the algorithm abide by the B-sets in the sense of Definition 4.5, so, since $x <_o w$, we deduce that x belongs to a smaller B-set than $B(o, q)$. Since the DAG does not contain interfering v-structures, the B-sets are ordered by inclusion. Therefore, we must have $x \in B(o, q)$. □

4.3 Explicit characterization of the set of possible candidates

In Algorithm 1, if w is added to the partial order, then $pa(v \downarrow w)$ will be O_v^k. Moreover, the ability to compute $u_{w|O_v^k}$ without integration is a necessary and sufficient condition for the joint density to be computed without integration, by Lemma 3.1. Hence, we must start the process with a node from the smallest possible B-set. That is, we only incorporate a node from $B_i(v)$ if all nodes from $B_{i-1}(v)$ are already included in O_v^k. Then, the elements of the smallest B-set larger than O_v^k (denoted as $B(O_v^k)$) are added.

Definition 4.6. *The smallest B-set strictly larger than a partial order O_v^k with $k < \left|pa(v)\right|$ is denoted by $B(O_v^k)$. Thus, $B(O_v^k) := B_{\tilde{q}}(v)$ with*

$$\tilde{q} := \min\{q \in \{1, \dots, Q(v)+1\};\, O_v^k \subsetneq B_q(v))\}.$$

Remark that such a $\tilde{q}$ always exists, since by definition $B_{Q(v)+1}(v) = pa(v)$. Furthermore, note that that $B(O_v^{|pa(v)|})$ is not defined, since at that moment the complete order $<_V$ on $pa(v)$ has been determined already.

Remark 4.1. *In Definition 4.6, the set $B(O_v^{|pa(v)|})$ is not defined because it is not possible to find a strictly larger B-set than $O_v^{|pa(v)|}$. Furthermore, the largest B-set $B_{Q(v)+1}$ is equal to $pa(v)$, by definition. Consequently, we always have $O_v^k \subsetneq B(O_v^k)$.*

It is important to note that including a node $w \in B(O_v^k) \setminus O_v^k$ ensures that this order abides by the B-sets. Therefore, the only allowed additions to a partial order O_v^k are nodes in $B(O_v^k) \setminus O_v^k$ for which we can compute $u_{w|O_v^k}$ without integration. This yields the following result.

Lemma 4.3. *$PossCand(O_v^k) = \big\{w \in B(O_v^k) \setminus O_v^k$ such that $u_{w|O_v^k}$ can be computed without integration$\big\}$.*

Lemma 4.3 already narrows down the set of possible candidates by restricting the choice of w from the whole set of parents of v to the set $B(O_v^k) \setminus O_v^k$. Next we fully characterize the set of possible candidates. It can be divided into three subsets, dependending on the local structure of node w in the graph. The most elementary case is when $w \in B(O_v^k)$ is such that $\mathrm{dsep}\big(w, O_v^k \,\big|\, \emptyset\big)$. Here, we have $u_{w|O_v^k} = u_w$ which obviously does not require any integration. We will refer to these nodes as **possible candidates by independence**. Note that this is in particular the case when $O_v^k = \emptyset$, i.e. $k = 0$.

Assume now that w is not a possible candidate by independence. The goal is to find conditions so $u_{w|O_v^k}$ can be computed without integration. First the largest possible set of nodes in O_v^k will be removed using conditional independence. Let us define J to be the largest set $J \subset O_v^k$ such that $\mathrm{dsep}\big(w, J \,\big|\, O_v^k \setminus J\big)$. Then we have $u_{w|O_v^k} = u_{w|O_v^k \setminus J}$. Since w is not a possible candidate by independence, we have $O_v^k \setminus J \neq \emptyset$. $u_{w|O_v^k \setminus J}$ can be computed without integration if there exists a proper recursion. By Definition 3.4, such a recursion needs to start with a specified conditional copula of the form $C_{w,o|O_v^k \setminus (J \sqcup \{o\})}$ where o is a node in $O_v^k \setminus J$.

Since J maximal then this copula cannot be specified to be the independence copula. Therefore, there must be $w \to o$ or $o \to w$. This means that a copula $C_{w,o|O_v^k \setminus \{o\}} = C_{w,o|O_v^k \setminus (J \sqcup \{o\})}$ must be equal to

- $C_{w,o|pa(o\downarrow w)}$, if $w \to o$, or,
- $C_{o,w|pa(w\downarrow o)}$, if $o \to w$.

In the first case, we say that w is a **possible candidate by incoming arc**. In the second case, we say that w is a **possible candidate by outgoing arc**. Such copulas have already been specified in the PCBN as they concern w and o, which are both parents of node v, hence they appear earlier in the well order of nodes. These conditions can be rewritten using d-separation, giving an explicit characterization of the set of possible candidates. Note that w is a possible candidate by incoming arc if $w \to o$ and

$$pa(o \downarrow w) \subseteq O_v^k \text{ and } \mathrm{dsep}\big(w, O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big) \,\big|\, pa(o \downarrow w) \sqcup \{o\}\big). \tag{9}$$

Indeed, the condition $pa(o \downarrow w) \subseteq O_v^k$ is necessary by Remark 3.2. The second part of (9) ensures that

$$C_{w,o|O_v^k \setminus o} = C_{w,o|pa(o\downarrow w)},$$

precisely because the nodes that can be removed from the conditioning set due to d-separation of w given the remaining nodes $pa(o \downarrow w) \sqcup \{o\}$ are $O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}$.

Similary we get that w is a possible candidates by outgoing arc if $o \to w$ and

$$pa(w \downarrow o) \subseteq O_v^k \text{ and } \mathrm{dsep}\big(w, O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\}) \,\big|\, pa(w \downarrow o) \sqcup \{o\}\big). \tag{10}$$

The second part of (10) ensures that

$$C_{o,w|O_v^k \setminus o} = C_{o,w|pa(w\downarrow o)},$$

as nodes in $O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\})$ can be removed from the conditioning set due to the d-separation.

Note that we have just proved the following result.

Proposition 4.1. *The set of possible candidates is composed*

$$PossCand(O_v^k) = PossCandInd(O_v^k) \sqcup PossCandIn(O_v^k) \sqcup PossCandOut(O_v^k),$$

where the three disjoint sets are defined as

$$\begin{aligned} PossCandInd(O_v^k) &:= \big\{w \in B(O_v^k) \setminus O_v^k;\ \mathrm{dsep}\big(w, O_v^k \,\big|\, \emptyset\big)\big\},\\ PossCandIn(O_v^k) &:= \big\{w \in B(O_v^k) \setminus O_v^k;\ \exists o \in O_v^k;\ w \to o \text{ and } (9)\big\},\\ PossCandOut(O_v^k) &:= \big\{w \in B(O_v^k) \setminus O_v^k;\ \exists o \in O_v^k;\ o \to w \text{ and } (10)\big\}. \end{aligned}$$

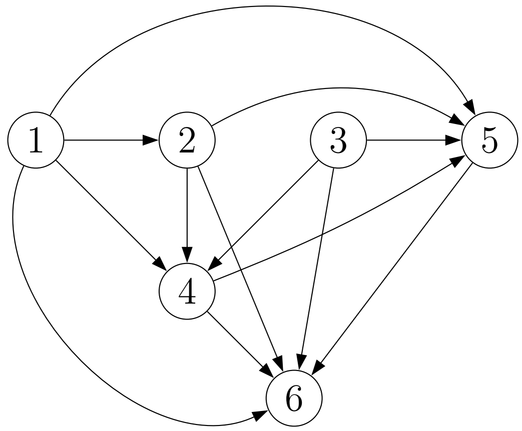
This proposition gives an explicit way of finding all possible candidates by looping over all nodes in $B(O_v^k)$ and testing whether they satisfy one of the conditions to be a possible candidates. This algorithm is implemented in the function `possible_candidates` of the `R` package `PCBN` [6].

In the example in the next section we illustrate the process of finding the parental orders for a specific DAG.

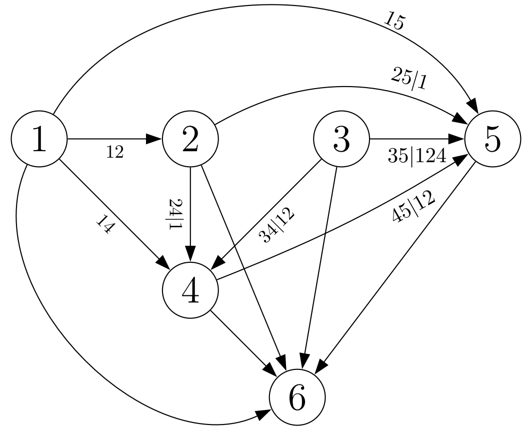
4.4 Example

Let us consider the graph in Figure 8 a). We choose to use the well-order $(1, 2, 3, 4, 5, 6)$ of nodes in this graph; note that it is not unique. The process for node $1, 2$ and 3 is simple as 1 and 3 have no parents and 2 has just one parent. We start our consideration at node 4.

1. Finding $<_4$. To find the parental order at 4 the algorithm is initiated with $O_4^0 = \emptyset$. Since node 4 has two children the B-sets are computed and we get that $B(4, 5) = \{1, 2, 3\}$ and $B(4, 6) = \{1, 2, 3\}$. This means that we have just one $B_1(4) = \{1, 2, 3\}$. Any of these nodes can be added as first to O_4^0. Since $B(O_4^0) \setminus O_4^0 = \{1, 2, 3\} \setminus \emptyset = \{1, 2, 3\}$ and by convention $\mathrm{dsep}\big(1, O_4^0 \,\big|\, \emptyset\big)$, $\mathrm{dsep}\big(2, O_4^0 \,\big|\, \emptyset\big)$ and $\mathrm{dsep}\big(3, O_4^0 \,\big|\, \emptyset\big)$ each of these nodes is a possible candidate by independence. If we choose the node 1, then $O_4^1 = (1)$.



(a) Initial graph.



(b) Graph with parental orders $1 <_4 2 <_4 3$ and $1 <_5 2 <_5 4 <_5 3$.

Figure 8: DAG on six nodes without active cycles and interfering v-structures.

Now $B(O_4^1) \setminus O_4^1 = \{1, 2, 3\} \setminus \{1\} = \{2, 3\}$. Since there is an arc $1 \to 2$ and the copula $C_{1,2|pa(4\downarrow 1)} = C_{1,2}$ is specified then node 2 is a possible candidate by incoming arc. Since $\mathrm{dsep}\big(3, O_4^1 \,\big|\, \emptyset\big)$ then the node 3 is a possible candidate by independence. If we choose node 2 and fix $O_4^2 = (1, 2)$, the last choice is $O_4^3 = (1, 2, 3)$, since as before the node 3 is a possible candidate by independence.

2. Finding $<_6$. Let us now assume that we have followed the process and fixed the orders $<_4$ and $<_5$ to be: $1 <_4 2 <_4 3$ and $1 <_5 2 <_5 4 <_5 3$. The graph together with the parental orderings for nodes 4 and 5 is presented in Figure 8 b). Our objective now is to choose a suitable ordering $<_6$ by growing a partial order O_6^k. Note that node 6 does not have any corresponding B-sets, as it has no children.

2.1. Finding O_6^1. Any node in $pa(6)$ can be added to $O_6^0 = \emptyset$ by independence. Suppose that we choose node 4 and get $O_6^1 = (4)$.

2.2. Finding O_6^2. None of the nodes in $B(O_6^1) \setminus O_6^1 = \{1, 2, 3, 5\}$ are d-separated from 4 by the empty set. Therefore, we must use one of the copulas corresponding to an arc connected to 4. The only suitable arc is the incoming arc $1 \to 4$, since its corresponding copula $C_{1,4}$ allows computation of the margin $u_{1|O_6^1} = u_{1|4}$.

2.3. Finding O_6^3. Next, we consider as possible candidates nodes in the set $B(O_6^2) \setminus O_6^2 = \{1, 2, 3, 4, 5\} \setminus \{4, 1\} = \{2, 3, 5\}$. There are two incoming arcs to node 4, i.e. $2 \to 4$ and $3 \to 4$. Node 2 is a possible candidate since the margin $u_{2|O_6^2} = u_{2|14}$ can be computed with the copula $C_{2,4|pa(4\downarrow 2)} = C_{2,4|1}$.

Node 3 is not a possible candidate by incoming arc $3 \to 4$. The copula corresponding to this arc, $C_{3,4|pa(4\downarrow 3)} = C_{3,4|1,2}$, contains node 2 in its conditioning set whereas $2 \notin O_6^2 = \{1, 4\}$. Hence, computing $u_{3|O_6^2} = u_{3|14}$ from the copula $C_{3,4|1,2}$ requires integration with respect to node 2:

$$u_{3|14} = \int_0^1 \frac{\partial C_{3,4|1,2}\big(u_3, u_4 \,|\, u_1, w_2\big)}{\partial u_4} \, \mathrm{d}w_2.$$

Note that $pa(4 \downarrow 3) = \{z \in pa(4); z <_4 3\} = \{1, 2\} \nsubseteq \{1, 4\} = O_6^2$ and this condition is necessary for a node w to be considered as a possible candidate for O_v^k by incoming arc $w \to o$. It must be that $pa(o \downarrow w) \subseteq O_v^k$, hence that parents of node o earlier in the ordering than w must have already been included in O_v^k. This condition, however, is not sufficient (as shown below). Indeed, if O_v^k contains nodes that are not in $pa(o \downarrow w)$, then these elements should be "removable" from the conditioning set of a copula by d-separation. Hence, another required condition is $\mathrm{dsep}\big(w, O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big) \,\big|\, pa(o \downarrow w) \sqcup \{o\}\big)$. We will examine this condition more closely below.

In this example, node 2 is the only node satisfying both of these conditions. Indeed, for $w = 2$ and $o = 4$, we have

$$\mathrm{dsep}\big(w, O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big) \,\big|\, pa(o \downarrow w) \sqcup \{o\}\big) = \mathrm{dsep}\big(2, \{1, 4\} \setminus \{1, 4\} \,\big|\, \{1, 4\}\big) = \mathrm{dsep}\big(2, \emptyset \,\big|\, \{1, 4\}\big)$$

which is satisfied by convention. Therefore, we add it to the partial order O_6^2, and obtain $O_6^3 = (4, 1, 2)$ and move on to the next iteration of the algorithm.

2.4. Finding O_6^4. Now we have two options: node 3 by incoming arc $3 \to 4$ and node 5 by the outgoing arc $4 \to 5$. Let us consider first the incoming arc $3 \to 4$. The condition $pa(4 \downarrow 3) = \{1, 2\} \subseteq O_6^3 = \{4, 1, 2\}$ is satisfied. Indeed, we can compute the conditional margin $u_{3|O_6^3} = u_{3|1,2,4}$ with the copula $C_{3,4|pa(4\downarrow 3)} = C_{3,4|1,2}$. Remark that the second condition is also satisfied since

$$\begin{aligned}\mathrm{dsep}\big(3, O_6^3 \setminus \big(pa(4 \downarrow 3) \sqcup \{4\}\big) \,\big|\, pa(4 \downarrow 3) \sqcup \{4\}\big) &= \mathrm{dsep}\big(3, \{1, 2, 4\} \setminus \{1, 2, 4\} \,\big|\, \{1, 2, 4\}\big) \\ &= \mathrm{dsep}\big(3, \emptyset \,\big|\, \{1, 2, 4\}\big)\end{aligned}$$

holds by convention. Hence, node 3 is a possible candidate by incoming arc $3 \to 4$ at this iteration. However, rather than adding node 3, we add node 5 by the outgoing arc $4 \to 5$. The conditions are satisfied:

$$pa(5 \downarrow 4) = \{1, 2\} \subseteq O_6^3 = \{4, 1, 2\} \quad \text{and}$$

$$\begin{aligned}\mathrm{dsep}\big(5, O_6^3 \setminus \big(pa(5 \downarrow 4) \sqcup \{4\}\big) \,\big|\, pa(5 \downarrow 4) \sqcup \{4\}\big) &= \mathrm{dsep}\big(5, \{1, 2, 4\} \setminus \{1, 2, 4\} \,\big|\, \{1, 2, 4\}\big) \\ &= \mathrm{dsep}\big(5, \emptyset \,\big|\, \{1, 2, 4\}\big)\end{aligned}$$

holds by convention.

2.5. Finding O_6^5. After the addition of node 5, the only remaining node is node 3. There are two incoming arcs $3 \to 5$ and $3 \to 4$. Note that in the previous iteration, it was possible to use the incoming arc $3 \to 4$. We now explain why we cannot use the incoming arc $3 \to 4$ anymore, even though the condition $pa(4 \downarrow 3) = \{1, 2\} \subseteq \{1, 2, 4, 5\} = O_6^4$ is satisfied. Note that we need the conditional margin $u_{3|O_6^4} = u_{3|1,2,4,5}$ but the arc $3 \to 4$ corresponds to the copula $C_{3,4|pa(4\downarrow 3)} = C_{3,4|1,2}$ from which $u_{3|1,2,4}$ can be computed. To remove node 5 from the conditioning set in $u_{3|1,2,4,5}$ the d-separation is required

$$\mathrm{dsep}\big(3, 5 \,\big|\, \{1, 2, 4\}\big) = \mathrm{dsep}\Big(3, O_6^4 \setminus \big(pa(4 \downarrow 3) \sqcup \{4\}\big) \,\Big|\, pa(4 \downarrow 3) \sqcup \{4\}\Big)$$

which does not hold due to the arc $3 \to 5$. The node 3 is a possible candidate by incoming arc $3 \to 5$ as it satisfies

$$\mathrm{dsep}\Big(3, \{1, 2, 4, 5\} \setminus \{1, 2, 4, 5\} \,\Big|\, \{1, 2, 4, 5\}\Big) = \mathrm{dsep}\Big(3, \emptyset \,\Big|\, \{1, 2, 4, 5\}\Big).$$

and

$$pa(5 \downarrow 3) = \{1,2,4\} \subseteq \{1,2,4,5\} = O_6^4.$$

Therefore, we can add node 3 by incoming arc $3 \to 5$ to obtain $O_6^5 = (4,1,2,5,3)$, giving us the order $4 <_6 1 <_6 2 <_6 5 <_6 3$.

5 Proof of Theorem 4.3

In this section, we prove Theorem 4.3. We need to prove that for every node v, at each step of our algorithm with current partial order O_v^k with $k < \left|pa(v)\right|$, the following property hold: $PossCand(O_v^k) \neq \emptyset$.

We will assume that the arguments of copulas assigned to arcs of the BN up to the current point of the algorithm (copulas assigned to arcs pointing to a node earlier in the well-ordering than node v and copulas assigned to arcs from nodes in O_v^k to v) do not require integration. This means that the following copulas have been assigned by our algorithm upon the arrival at O_v^k:

- $C_{x,y|pa(y\downarrow x)}$ with $x \to y \in E$ and $y < v$.
- $C_{o_j,v|O_v^{j-1}}$ with $j \leq k$.

The proof requires many additional results regarding properties of trails, B-sets, partial orders and possible candidates. These results can be found in the Appendix.

We will show that at any point of the algorithm, we are able to extend the current order O_v^k with a node $w \in PossCand(O_v^k)$. Thus, we must prove that there exists a node $w \in PossCand(O_v^k)$.

If $PossCandInd(O_v^k)$ is not empty, then the proof is complete. Therefore, in the rest of the proof we assume that $PossCandInd(O_v^k) = \emptyset$. Consequently, we can apply Lemma B.2 to find that $ad(O_v^k) \cap B(O_v^k) \neq \emptyset$. Thus, there exist a $w_1 \in B(O_v^k) \setminus O_v^k$ and $o_1 \in O_v^k$ such that $w_1 \to o_1$ or $o_1 \to w_1$.

In what follows we will show that the existence of an arc $w_1 \to o_1$ implies that $PossCandIn(O_v^k)$ is not empty. Subsequently we will assume that no arc of the form $w_1 \to o_1$ exists and we prove that this together with the existence of an arc $o_1 \to w_1$ implies that $PossCandOut(O_v^k)$ is not empty, concluding the proof. The cases of the existence of the arcs $w_1 \to o_1$ and $o_1 \to w_1$ are considered separately.

5.1 First case: $w_1 \to o_1$

If w_1 can be added to O_v^k by the incoming arc $w_1 \to o_1$, then $w_1 \in PossCandIn(O_v^k)$, completing the proof. Thus, we assume that $w_1 \notin PossCandIn(O_v^k)$, for every w_1 which is a parent of o_1 and which belongs to $B(O_v^k) \setminus O_v^k$. Formally, this means

$$pa(o_1) \cap \left(B(O_v^k) \setminus O_v^k\right) \cap PossCandIn(O_v^k) = \emptyset. \tag{11}$$

To get a contradiction our strategy is to apply the lemma below. This lemma states that under the assumptions above, the arc $w_1 \to o_1$ implies the existence of another pair of nodes $w_2 \in B(O_v^k) \setminus O_v^k$ and $o_2 \in O_v^k$ such that $w_2 \to o_2$ and $o_1 \to o_2$. It is the case that $o_2 \neq o_1$, but w_1 and w_2 may be the same node.

Lemma 5.1. *Assume that there exist $w_1 \in B(O_v^k) \setminus O_v^k$ and $o_1 \in O_v^k$ such that $w_1 \to o_1$. If (11) holds, then there exist $w_2 \in B(O_v^k) \setminus O_v^k$ and $o_2 \in O_v^k$ such that $w_2 \to o_2$ and $o_1 \to o_2$.*

Applying Lemma 5.1 iteratively, we obtain an infinite sequence $o_1 \to o_2 \to o_3 \to \cdots$ of connected nodes of G. Since the graph G is acyclic and has a finite number of nodes, such a sequence cannot exist. Therefore (11) cannot be true, which means that $PossCandIn(O_v^k) \neq \emptyset$, completing the proof. It remains to prove Lemma 5.1.

5.1.1 Proof of Lemma 5.1

Without loss of generality, we can assume that w_1 is the smallest element with respect to $<_{o_1}$ in $pa(o_1) \cap \big(B(O_v^k) \setminus O_v^k\big)$. From (11), we know that $w_1 \notin PossCandIn(O_v^k)$, hence (at least) one of the two following restrictions must be violated:

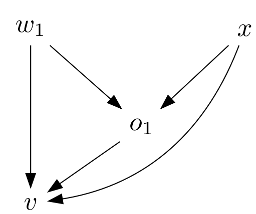
1. $pa(o_1 \downarrow w_1) \subseteq O_v^k$,
2. $\operatorname{dsep}\big(w_1, O_v^k \setminus \big(pa(o_1 \downarrow w_1) \sqcup \{o_1\}\big) \,\big|\, pa(o_1 \downarrow w_1) \sqcup \{o_1\}\big)$.

The first restriction is satisfied by the lemma below.

Lemma 5.2. *Let w_1 be the smallest element in $B(O_v^k) \cap pa(o_1)$ with respect to $<_{o_1}$. Then, $pa(o_1 \downarrow w_1) \subseteq O_v^k$.*

Proof of Lemma 5.2. Suppose that there exists an $x \in pa(o_1 \downarrow w_1) \setminus O_v^k$. That is, $x \in pa(o_1) \setminus O_v^k$ and $x <_{o_1} w_1$. Since w_1 is the smallest node in $B(O_v^k) \cap pa(o_1)$, we have $x \notin B(O_v^k)$.

Remark that $B(o_1, v) = pa(o_1) \cap pa(v)$. Since $w_1 \to v$ and $w_1 \to o_1$, we know that $w_1 \in B(o_1, v)$. Since $x <_{o_1} w_1$, by Corollary 4.2, we obtain $x \to v$. This mean that G contains the subgraph below.



Let q be a node such that $B(O_v^k) = B(v, q)$. Then we must have $v \to q$ and $w_1 \to q$. Since the B-sets are ordered by inclusion and $o_1 \in O_v^k \subset B(O_v^k) = B(v, q)$, we get that $o_1 \to q$. So $w_1 \in B(o_1, q)$. By Corollary 4.2, we obtain $x \to q$. So $x \in B(o_1, q) = B(O_v^k)$, which is a contradiction. $\square$

Therefore, the second restriction must be violated. We consider two possible cases. Assume that $pa(o_1 \uparrow w_1) \cap O_v^k \neq \emptyset$. Lemma B.5 immediately implies that there exists an $o_2 \in O_v^k$ as desired (where w_1 is w, o_1 is o_i and o_2 is $\tilde{o}$ in the notation of Lemma B.5) and we set $w_2 := w_1$, completing the proof in this case.

Assume now that $pa(o_1 \uparrow w_1) \cap O_v^k = \emptyset$. We can apply Lemma C.1 (with $w = w_1$ and $o = o_1$ in the notation of Lemma C.1) to find that there exists a trail between w_1 and a node $\tilde{o} \in O_v^k \setminus \big(pa(o_1 \downarrow w_1) \sqcup \{o_1\}\big)$

which is activated by $pa(o_1 \downarrow w_1) \sqcup \{o_1\}$ containing no converging connections. Let us pick a shortest such trail.

$$w_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons \tilde{o}.$$

Remark that this trail is a shortest trail activated by the empty set between $w_1 \in B(O_v^k)$ and $\tilde{o} \in O_v^k \subseteq B(O_v^k)$ consisting of nodes in $V \setminus \big(pa(o_1 \downarrow w_1) \sqcup \{o_1\}\big)$. Therefore, we can combine Lemma A.1 (with $K = V \setminus \big(pa(o_1 \downarrow w_1) \sqcup \{o_1\}\big)$) and Lemma B.1 to find that $\{x_1, \ldots, x_n\} \subseteq B(O_v^k)$. In particular we obtain $x_n \in B(O_v^k)$.

Furthermore, x_n cannot be contained in O_v^k. Otherwise, the trail from w to x_n would be an even shorter active trail from w to a node in O_v^k. Hence, $x_n \in B(O_v^k) \setminus O_v^k$.

Now, the trail

$$o_1 \leftarrow w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons \tilde{o}$$

is an active trail between two nodes in O_v^k given the empty set consisting of nodes not in O_v^k. Thus, by Lemma B.3, o_1 and $\tilde{o}$ must be adjacent. By the assumption that $pa(o_1 \uparrow w_1) \cap O_v^k = \emptyset$, we have $\tilde{o} \notin pa(o_1 \uparrow w_1)$. Remark that $\tilde{o} \notin pa(o_1 \downarrow w_1) \sqcup \{o_1\}$ (by definition of $\tilde{o}$) and that $pa(o_1) = pa(o_1 \downarrow w_1) \sqcup pa(o_1 \uparrow w_1) \sqcup \{w_1\}$. This shows that $\tilde{o} \notin pa(o_1)$. Therefore, we must have $o_1 \in pa(\tilde{o})$; this means that we have the subgraph below.

$$\begin{tikzcd}
o_1 \arrow[rrrrrr, bend left] & w_1 \arrow[l] \arrow[r, phantom, "\rightleftharpoons"] & x_1 \arrow[r, phantom, "\rightleftharpoons"] & x_2 \arrow[r, dotted, no head] & x_{n-1} \arrow[r, phantom, "\rightleftharpoons"] & x_n \arrow[r] & \tilde{o}
\end{tikzcd}$$

Clearly, $\tilde{o}$ is our desired node o_2 and x_n is our desired node w_2. Indeed, we have $o_1 \to \tilde{o}$ and $x_n \to \tilde{o}$, with $\tilde{o} \in O_v^k$ and $x_n \in B(O_v^k) \setminus O_v^k$.

5.2 Second case: $o_1 \to w_1$

First, we remark that if E contains arcs of the form $w \to o$ with $w \in B(O_v^k) \setminus O_v^k$ and $o \in O_v^k$, then by the previous case we have that $PossCandIn(O_v^k) \neq \emptyset$. Therefore, we can assume without loss of generality that there are no such arcs in E.

If $w_1 \in PossCandOut(O_v^k)$, then the proof is complete. Assume now that $w_1 \notin PossCandOut(O_v^k)$. We will show that $PossCandOut(O_v^k) \neq \emptyset$ with the lemma below. The lemma states that under the assumptions above, the arc $o_1 \to w_1$ implies the existence of another pair of nodes $w_2 \in B(O_v^k) \setminus O_v^k$ and $o_2 \in O_v^k$ such that w_1 and w_2 are connected by a trail where all arcs point in direction of w_1. We can repeat this argument and construct a sequence of nodes.

Lemma 5.3. *Let $w_1 \in B(O_v^k) \setminus O_v^k$ and $o_1 \in O_v^k$ such that $o_1 \to w_1 \in E$. Assume that $w_1 \notin PossCandOut(O_v^k)$, and that*

$$E \text{ does not contain any arc from } B(O_v^k) \text{ to } O_v^k. \tag{12}$$

Then, there exist $w_2 \in B(O_v^k) \setminus O_v^k$ and $o_2 \in O_v^k$ such that $o_2 \to w_2$, and w_1 and w_2 are connected by a trail of the form

$$w_1 \leftarrow x_1 \leftarrow \cdots \leftarrow x_n \leftarrow w_2.$$

Applying Lemma 5.3 iteratively, we obtain a sequence

$$w_1 \leftarrow \cdots \leftarrow w_2 \leftarrow \cdots \leftarrow w_3 \leftarrow \cdots .$$

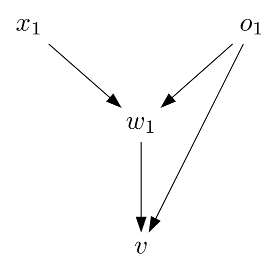
Since the graph G is acyclic and has a finite set of nodes, this sequence must be finite. Let w^* be the last element of the longest sequence that can be constructed starting from w_1. Then, w^* must belong to $PossCandOut(O_v^k)$, otherwise, it would not be the last. Hence, we have $PossCandOut(O_v^k) \neq \emptyset$, completing the proof. It remains to prove Lemma 5.3.

5.2.1 Proof of Lemma 5.3

Without loss of generality, we can assume that o_1 is the largest element in $O_v^k \cap pa(w_1)$ with respect to $<_{w_1}$. We consider two cases.

First case: $pa(w_1 \downarrow o_1) \setminus O_v^k \neq \emptyset$.

Let $x_1 \in pa(w_1 \downarrow o_1) \setminus O_v^k$. That is, $x_1 \in pa(w_1) \setminus O_v^k$ and $x_1 <_{w_1} o_1$. This means that G contains the subgraph below.



Note that $o_1 \in pa(w_1) \cap pa(v) = B(w_1, v)$, with $x_1 <_{w_1} o_1$ By Corollary 4.2, we get $v \to x_1$.

Let q be a node such that $B(O_v^k) = B(v, q)$. Then we know that $v \to q$, $w_1 \to q$ and $o_1 \to q$. Therefore, $x_1 \in pa(w_1) \cap pa(v) = B(w_1, q)$, with $x_1 <_{w_1} o_1$. By Corollary 4.2, we get $q \to x_1$.

We have shown that $x_1 \in pa(v)$ and $x_1 \in pa(q)$, so $x_1 \in B(v, q) = B(O_v^k)$.

By the assumption that $PossCandInd(O_v^k) = \emptyset$, we have $\cancel{\text{dsep}}\big(x_1, O_v^k \,\big|\, \emptyset\big)$. Let us pick a shortest trail from x_1 to O_v^k

$$x_1 \rightleftharpoons x_2 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons \tilde{o} \tag{13}$$

activated by the empty set with $\tilde{o} \in O_v^k$. Note that x_1 and $\tilde{o}$ are both included in the B-set $B(O_v^k)$. Therefore, by Lemma B.1, we have that $x_i \in B(O_v^k)$ for all $i = 1, \ldots, n$. In particular, $x_n \in B(O_v^k)$. Furthermore, x_n does not belong to the set O_v^k. Otherwise, the trail

$$x_1 \rightleftharpoons x_2 \rightleftharpoons \cdots \rightleftharpoons x_n$$

would be a shorter trail from x_1 to O_v^k activated by the empty set than (13), which is a contradiction. Therefore, x_n must be in $B(O_v^k) \setminus O_v^k$.

By (12), we know that there is no arc pointing from a node in $B(O_v^k) \setminus O_v^k$ to a node in O_v^k. This means that the arc $x_n \to \tilde{o}$ is not possible. Consequently, the trail (13) must contain the arc $x_n \leftarrow \tilde{o}$. Since the trail is activated by the empty set it contains no converging connections by Lemma A.2. Therefore, (13) must be of the form $w_1 \leftarrow x_1 \leftarrow \cdots \leftarrow x_n \leftarrow \tilde{o}$ with $x_n \in B(O_v^k) \setminus O_v^k$ and $\tilde{o} \in O_v^k$. Hence, x_n is our desired node w_2 and $\tilde{o}$ is our desired node o_2.

Second case: $pa(w_1 \downarrow o_1) \setminus O_v^k = \emptyset$.

By assumption, we have $w_1 \notin \mathit{PossCandOut}(O_v^k)$. Since $pa(w_1 \downarrow o_1) \setminus O_v^k = \emptyset$, we must have $pa(w_1 \downarrow o_1) \subseteq O_v^k$. Therefore, using the definition of $\mathit{PossCandOut}(O_v^k)$ (see Equation (10)), we must have

$$\cancel{\mathrm{dsep}}\big(w_1, O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\}) \,\big|\, pa(w_1 \downarrow o_1) \sqcup \{o_1\}\big).$$

Therefore, there exists a trail between w_1 and a node in $O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\})$ activated by $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$. By Lemma C.3, there exists such a trail containing no converging connections. Thus, we can pick a shortest trail from w to $O_v^k \setminus pa(w_1 \downarrow o_1) \sqcup \{o_1\}$ activated by $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$ containing no converging connections:

$$w_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons \tilde{o} \tag{14}$$

with $\tilde{o} \in O_v^k \setminus pa(w_1 \downarrow o_1) \sqcup \{o_1\}$.

First, we show that (14) must be of length $n > 1$. If $n = 0$, then we would have that $w_1 \rightleftharpoons \tilde{o}$. This arc must point to the left, since the arc $w_1 \to \tilde{o}$ is an arc from a node in $B(O_v^k) \setminus O_v^k$ to a node in O_v^k which cannot be present by Condition (12). Because there is the arc $w_1 \leftarrow \tilde{o}$, we know that $\tilde{o} \in pa(w_1)$. Moreover, by definition the node $\tilde{o}$ does not belong to $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$, and thus $\tilde{o} \in pa(w_1 \uparrow o_1)$. This means that $o_1 <_{w_1} \tilde{o}$. Since we picked o_1 to be largest element in $O_v^k \cap pa(w_1)$ according to $<_{w_1}$ then $o_1 <_{w_1} \tilde{o}$ is not possible, proving that $n > 1$.

Now, we show that for all $i = 1, \ldots, n$, $x_i \notin O_v^k$. Suppose that for some i, we have that $x_i \in O_v^k$. The set O_v^k can be rewritten as $O_v^k = \big(O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\})\big) \sqcup (pa(w_1 \downarrow o_1) \sqcup \{o_1\})$. Since $x_i \in O_v^k$, it must be in $O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\})$ or in $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$. If $x_i \in O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\})$, then the $w_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_i$ would be a shorter trail between w_1 and $O_v^k \setminus (pa(w_1 \downarrow o_1) \sqcup \{o_1\})$ activated by $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$ than trail (14). This is a contradiction, because we picked the shortest such trail. Hence, x_i must be in $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$. However, in this case the trail (14) would be blocked by $pa(w_1 \downarrow o_1) \sqcup \{o_1\}$ which is also a contradiction. Therefore, x_i cannot be in O_v^k proving the claim.

The trail (14) is the shortest trail activated by the empty set between two nodes in $B(O_v^k)$ (w and $\tilde{o}$), and thus by Lemma B.1, $x_i \in B(O_v^k)$ for all $i = 1, \ldots, n$. This means that for all $i = 1, \ldots, n$, $x_i \in B(O_v^k) \setminus O_v^k$, in particular $x_n \in B(O_v^k) \setminus O_v^k$. Therefore, the arrow $x_n \to \tilde{o}$ is forbidden by Condition (12) and so we must have $x_n \leftarrow \tilde{o}$. As a consequence, using the fact that (14) has no converging connection, we deduce that (14) takes the form

$$w_1 \leftarrow x_1 \leftarrow \cdots \leftarrow x_n \leftarrow \tilde{o}$$

with $x_n \in B(O_v^k) \setminus O_v^k$ and $\tilde{o} \in O_v^k$ and $n > 0$. So, x_n is our desired node w_2 and $\tilde{o}$ is our desired node o_2.

This completes the proof of Lemma 5.1. The proof of Theorem 4.3 is completed.

6 Parameter estimation of PCBN models via estimating equations

6.1 Methodology and results

In this section, we focus on the estimation of the (conditional) copulas in PCBN model. Indeed, estimation of the marginal densities can be done using classical univariate techniques (e.g. using parametric methods such as maximum likelihood or method of moments, or non-parametric methods such as kernel smoothing).

We will assume that the marginal distributions have been estimated nonparametrically via ranking. The results can be adapted in the same way for parametric margins.

Due to the product structure of the joint copula density in PCBNs it is necessary to estimate $c_{wv|pa(v\downarrow w)}$, for each $v \in V$ and $w \in pa(v)$. We focus on parametric conditional copulas which are of the form

$$c_{w,v|pa(v\downarrow w)}\big(u_{w|pa(v\downarrow w)}\,,\, u_{v|pa(v\downarrow w)}\,\big|\, u_{pa(v\downarrow w)}\big) = c_{\boldsymbol{\theta}_{w\to v}}\big(u_{w|pa(v\downarrow w)}\,,\, u_{v|pa(v\downarrow w)}\,\big|\, u_{pa(v\downarrow w)}\big)$$

where $u_{w|pa(v\downarrow w)} \in [0,1]$, $u_{v|pa(v\downarrow w)} \in [0,1]$, $u_{pa(v\downarrow w)} \in [0,1]^{|pa(v\downarrow w)|}$, and $\boldsymbol{\theta}_{w\to v} \in \boldsymbol{\Theta}_{w\to v}$, for a given parametric family of (conditional) copulas $\mathscr{C}_{w\to v} = \{c_\theta;\, \theta \in \boldsymbol{\Theta}_{w\to v}\}$.

By definition, $c_{wv|pa(v\downarrow w)}$ is the conditional copula of $u_{w|pa(v\downarrow w)}$ and $u_{v|pa(v\downarrow w)}$ given $u_{pa(v\downarrow w)}$. From Theorem 4.1, we know that a PCBN (G, O) with neither active cycles nor interfering v-structures does not necessitate integration to compute these conditional margins. Moreover, we have shown that to prevent integration the assignment of copulas, O, must be determined by Algorithm 1. Therefore, we impose both restrictions on the class of PCBNs.

Definition 6.1 (PCBN model)**.** *Let (G, O) be a PCBN. Let $\mathscr{C}_{w\to v} = \big\{c_{\boldsymbol{\theta}_{w\to v}};\, \boldsymbol{\theta}_{w\to v} \in \boldsymbol{\Theta}_{w\to v}\big\}$ be a collection of bivariate (conditional) pair-copula densities for each arc $w \to v \in E$. Let $\boldsymbol{\Theta} := \times_{v\in V} \times_{w\in pa(v)} \boldsymbol{\Theta}_{w\to v}$, and for $\boldsymbol{\theta} = (\boldsymbol{\theta}_{w\to v})_{v\in V, w\in pa(v)} \in \boldsymbol{\Theta}$, let*

$$c_{\boldsymbol{\theta}}(\mathbf{u}_V) := \prod_{v\in V} \prod_{w\in pa(v)} c_{\boldsymbol{\theta}_{w\to v}}\big(u_{w|pa(v\downarrow w)}\,,\, u_{v|pa(v\downarrow w)}\,\big|\, u_{pa(v\downarrow w)}\big), \tag{15}$$

*where $u_{w|pa(v\downarrow w)}$, $u_{v|pa(v\downarrow w)}$, $u_{pa(v\downarrow w)}$ have been computed using the recursion of h-functions with the parameter $\boldsymbol{\theta}$. Then, the collection of densities $\mathscr{C} = \{c_{\boldsymbol{\theta}};\, \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$ is a **PCBN model** on $[0,1]^{|V|}$ corresponding to the PCBN (G, O). We say that this is a **restricted PCBN model** if (G, O) does not require integration.*

Let us assume that we observe $N > 1$ i.i.d. observations $\mathscr{D} = (\mathbf{X}^1, \dots, \mathbf{X}^N)$ from the density f_V, whose copula is assumed to belong to $\mathscr{C}$ with parameter $\boldsymbol{\theta}^* = (\boldsymbol{\theta}^*_{w\to v})_{v\in V,\, w\in pa(v)} \in \boldsymbol{\Theta}$. Let $\hat{\mathscr{D}} := (\hat{\mathbf{U}}^1, \dots, \hat{\mathbf{U}}^N)$ be the dataset after applying the marginal empirical cdfs component-wise.

The estimation problems for the PCBN model will be similar as the ones encountered for the estimation of vine copulas models [8]. A naive way to estimate $\boldsymbol{\theta}^*$ is by maximizing the pseudo-log-likelihood

$$\begin{aligned}\ell(\boldsymbol{\theta};\, \mathscr{D}) &= \log\left(\prod_{i=1}^{n} \prod_{v\in V} \prod_{w\in pa(v)} c_{\boldsymbol{\theta}_{w\to v}}\big(\hat{U}^i_{w|pa(v\downarrow w)}\,,\, \hat{U}^i_{v|pa(v\downarrow w)}\,\big|\, \hat{U}^i_{pa(v\downarrow w)}\big)\right) \\ &=: \sum_{v\in V} \sum_{w\in pa(v)} \ell_{w\to v}(\boldsymbol{\theta}_{w\to v};\, \hat{\mathscr{D}}_{w\to v}),\end{aligned}$$

where $\hat{\mathscr{D}}_{w\to v} := (\hat{U}^i_{w|pa(v\downarrow w)}, \hat{U}^i_{v|pa(v\downarrow w)}, \hat{U}^i_{pa(v\downarrow w)})_{i=1,\dots,N}$. This is difficult since $\hat{\mathscr{D}}_{w\to v}$ depends on $\boldsymbol{\theta}$, implicitly, via the recursion of h-functions. Therefore, as in the vine copula models [8], we propose to estimate $\boldsymbol{\theta}$ using a stepwise procedure: for each $v \in V$ and for every $w \in pa(v)$, we estimate the copula $c_{w,v|pa(v\downarrow w)}$ using the dataset $\hat{\mathscr{D}}_{w\to v}$. This dataset can be obtained easily, without integration in an iterative way assuming that the PCBN is restricted, and the stepwise procedure is done in the 'right' order, i.e. using a well-ordering on the node set V and the parental orderings O.

We follow [8, Section 3.1] and apply the framework presented in [18]. To construct convergent and asymptotically normal estimators for parameters of PCBN models we propose to use stepwise estimating equations. For every arc $w \to v$, let $\phi_{w\to v}$ be an $\mathbb{R}^{\dim(\boldsymbol{\Theta}_{w\to v})}$-valued function on $[0,1]^2 \times [0,1]^{|pa(v\downarrow w)|} \times \boldsymbol{\Theta}_{w\to v}$. We estimate $\boldsymbol{\theta}^*_{w\to v}$ by the rank approximate Z-estimator $\hat{\boldsymbol{\theta}}_{w\to v}$ defined as a solution of

$$\sum_{i=1}^{N} \phi_{w\to v}(\hat{U}^i_{w|pa(v\downarrow w)}, \hat{U}^i_{v|pa(v\downarrow w)}, \hat{U}^i_{pa(v\downarrow w)}, \boldsymbol{\theta}_{w\to v}) = 0. \tag{16}$$

where $\hat{\mathscr{D}}_{w\to v}$ depends on the previously estimated parameters. We get that there exists a function $\tilde{\phi}_{w\to v}$: $[0,1]^{|V|} \times \mathbb{R}^{\dim(\boldsymbol{\Theta})} \to \mathbb{R}^{\dim(\boldsymbol{\Theta}_{w\to v})}$ such that

$$\tilde{\phi}_{w\to v}(\hat{\mathbf{U}}^i,\, \boldsymbol{\theta}) = \phi_{w\to v}(\hat{U}^i_{w|pa(v\downarrow w)}, \hat{U}^i_{v|pa(v\downarrow w)}, \hat{U}^i_{pa(v\downarrow w)}, \boldsymbol{\theta}_{w\to v}).$$

Moreover, this function $\tilde{\phi}_{w\to v}$ only depends on the $\hat{U}^i_j$ for j earlier than v in the well-ordering and parental ordering up to w. This holds also for the components of $\boldsymbol{\theta}$. Therefore, the stepwise rank approximate Z-estimator $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}}_{w\to v})_{v\in V,\, w\in pa(v)}$ is the solution of the estimating equations

$$\sum_{i=1}^{N} \tilde{\phi}(\hat{U}^i_1, \dots, \hat{U}^i_n, \boldsymbol{\theta}) = 0,$$

where $\tilde{\phi}$ is the function obtained by concatenating all the outputs of $\tilde{\phi}_{w\to v}$, for $v \in V$ and $w \in pa(v)$. The estimation procedure is summarized in Algorithm 2.

Algorithm 2 Computation of the stepwise rank approximate Z-estimator of $\boldsymbol{\theta}$

Require: PCBN (G, O), PCBN model $\mathscr{C}$, data $\mathscr{D} = (\mathbf{X}^1, \dots, \mathbf{X}^N)$

Compute the pseudo-observations $\hat{\mathbf{U}}^1, \dots, \hat{\mathbf{U}}^N$

for each node v in V according to a well-ordering **do**

  for each node w in $pa(v)$ according the order $<_v$ **do**

    Compute the conditional margins $\hat{U}^i_{w|pa(v\downarrow w)}$ and $\hat{U}^i_{v|pa(v\downarrow w)}$, for $i = 1, \dots, N$, via the recursion of h-functions, using the previously estimated copula parameters.

    Compute $\hat{\boldsymbol{\theta}}_{w\to v}$ as the solution of the Estimating Equation (16).

  end for

end for

return $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}}_{w\to v})_{v\in V,\, w\in pa(v)}$

The following result is a direct application of Theorem 1 in [18].

Theorem 6.1. *Under classical conditions (A1)–(A5) in [18] on $\tilde{\phi}$, $\boldsymbol{\theta}^*$ and $\mathbb{P}$, there exists a positive definite matrix A of size $\dim(\Theta)^2$ such that $\sqrt{n}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)$ converges in distribution, as $N \to \infty$, to a multivariate normal distribution with mean 0 and covariance matrix A.*

Note that Theorem 6.1 also holds if the PCBN is not restricted. In this case, however, computation of the pseudo-observations becomes much more computationally costly, since integration on potentially high-dimensional spaces will be required.

Theorem 6.1 shows the consistency and asymptotic normality of the stepwise pseudo-maximum likelihood estimator, using the estimating functions

$$\phi_{w\to v}(\hat{\mathbf{U}}^i, \boldsymbol{\theta}_{w\to v}) = \nabla_{\boldsymbol{\theta}_{w\to v}} c_{\boldsymbol{\theta}_{w\to v}}\big(\hat{U}^i_{w|pa(v\downarrow w)}, \hat{U}^i_{v|pa(v\downarrow w)} \,\big|\, \hat{U}^i_{pa(v\downarrow w)}\big) / c_{\boldsymbol{\theta}_{w\to v}}\big(\hat{U}^i_{w|pa(v\downarrow w)}, \hat{U}^i_{v|pa(v\downarrow w)} \,\big|\, \hat{U}^i_{pa(v\downarrow w)}\big)$$

This holds under usual conditions for pseudo-maximum likelihood estimators (domination condition on the derivatives of $\tilde{\phi}$, integrability condition of $\tilde{\phi}$, identifiability of the model and existence of a nonsingular Fisher information matrix). Other estimation techniques are also included in this framework, such as estimation by inversion of Kendall's tau.

To get practical insights about the performance of these estimation techniques, a simulation example is presented in the next section.

6.2 Small simulation study

In this section, we show that Algorithm 2 can accurately estimate the parameters given a data set generated from a known PCBN. We study a particular PCBN with graphical structure and assignment of copulas as in Figure 9 and parameters from Table 1.

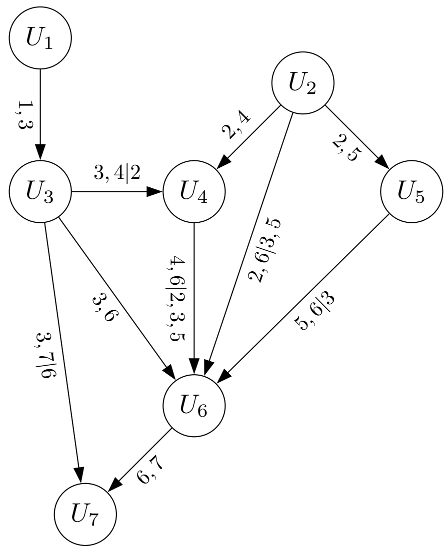


Figure 9: PCBN used for the simulation studies.

Arc	Copula	Family	Kendall's τ
$1 \to 3$	$c_{1,3}$	Gumbel	0.6
$2 \to 4$	$c_{2,4}$	Joe	0.8
$3 \to 4$	$c_{3,4\|2}$	Gumbel	0.6
$2 \to 5$	$c_{2,5}$	Frank	0.7
$3 \to 6$	$c_{3,6}$	Joe	0.9
$5 \to 6$	$c_{5,6\|3}$	Frank	0.6
$2 \to 6$	$c_{2,6\|3,5}$	Frank	0.85
$4 \to 6$	$c_{4,6\|2,3,5}$	Gumbel	0.75
$6 \to 7$	$c_{6,7}$	Gumbel	0.65
$3 \to 7$	$c_{3,7\|6}$	Joe	0.55

Table 1: The copula families and parameters of the PCBN in Figure 9.

We study the influence of the sample size n, with possible values $10, 20, 50, 80, 100, 200, 500, 800, 1000$ on the mean square error of the estimated parameters. Since the copula families that are studied are different, we reparametrize them by their Kendall's tau.

We study two possible estimation methods for the parameter of the copula: by maximum likelihood and by inversion of Kendall's tau. In both cases, we distinguish between known margins, and unknown margins, estimated non-parametrically via their rank (by the function *pobs()* from the *VineCopula* package [14]). Since this article does not focus on model selection, we assume that the graph structure, orders and copula families are known. Model selection for PCBN will be treated in a future work.

The mean-square error (MSE) of an estimator is then defined as the average squared difference between the true value of the parameter and its estimate. We do 100 replications to estimate the MSE. The estimation results are presented on Figure 10.

We can observe that our estimators are all converging (eventually) at the rate $1/\sqrt{n}$. But the time to reach this regime and the constant multiplicative factor can be very different.

Estimating unconditional copulas is relatively easy and the estimate converge fast to the true values. However, for larger conditioning sets we can see a slower convergence of the estimates. This happens even in the case where the simplifying assumption is made, both in the model specification and in the estimation procedure. The observation is in line with the recent results of [7] about the asymptotics of statistical models with diverging number p of parameters (meaning that $p = p_n \to \infty$). Indeed, when the number of parameters increases (i.e., here when we are adding arcs to our graph), the sample size necessary to reach a certain accuracy needs to be larger.

Interestingly, the knowledge of the margins seems to have a stronger influence on the MSE for the

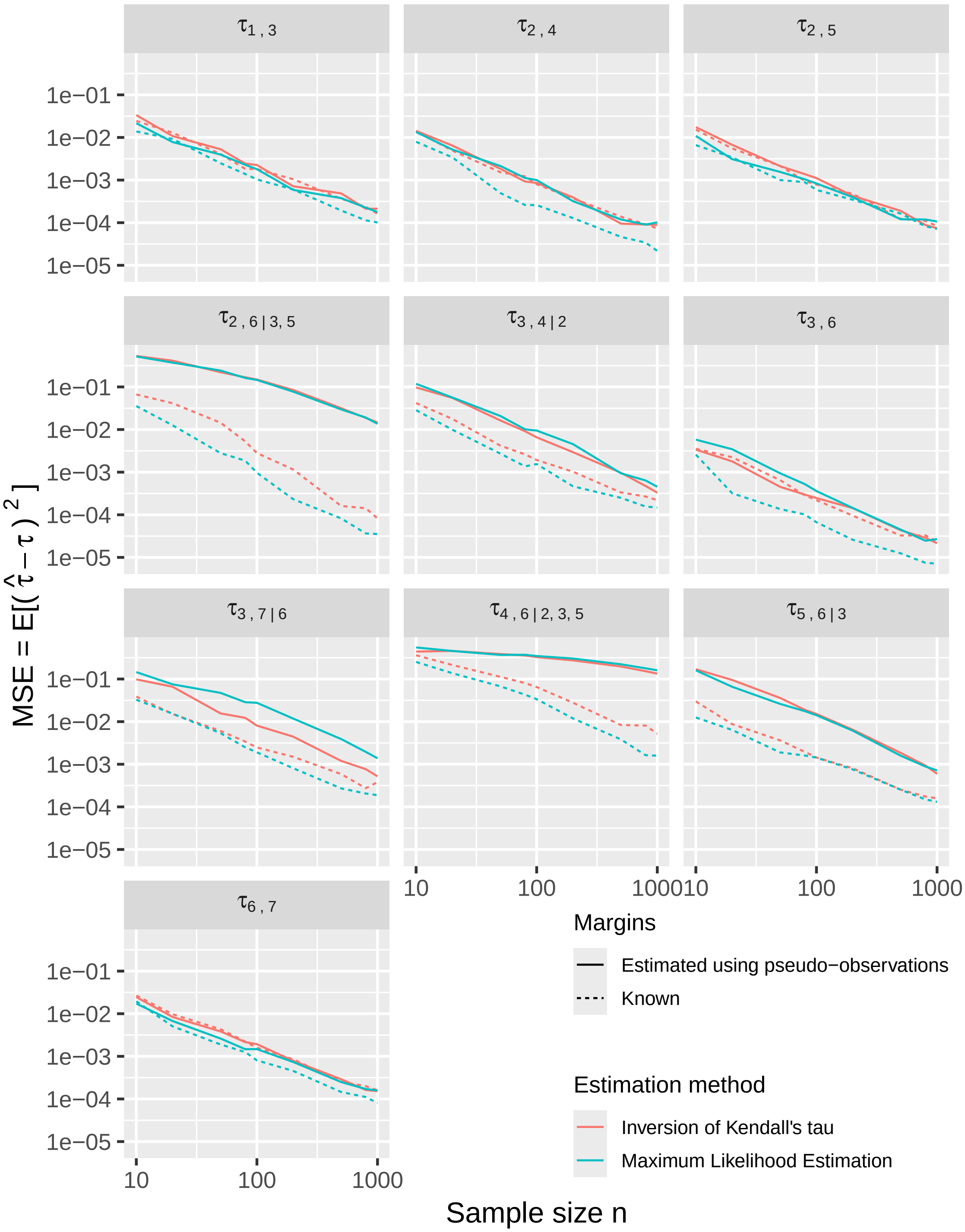


Figure 10: MSE as a function of the sample size, for different nodes of the PCBN.

estimation of the copulas with larger conditioning sets.

References

[1] A. Bauer and C. Czado. Pair-copula bayesian networks. *Journal of Computational and Graphical Statistics*, 25(4):1248–1271, 2016.

[2] A. Bauer, C. Czado, and T. Klein. Pair-copula constructions for non-gaussian dag models. *Canadian Journal of Statistics*, 40:86–109, 2012.

[3] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probabilistic Networks and Expert Systems: Exact Computational Methods for Bayesian Networks*. Information Science and Statistics. Springer New York, 1999.

[4] A. Derumigny and J.-D. Fermanian. About tests of the "simplifying" assumption for conditional copulas. *Dependence Modeling*, 5(1):154–197, 2017.

[5] A. Derumigny, N. Horsman, and D. Kurowicka. Minimal trails in restricted DAGs. *ArXiv preprint, arXiv:2510.02113*, 2025.

[6] A. Derumigny, N. Horsman, and D. Kurowicka. *PCBN: Inference of Pair-Copula Bayesian Networks*, 2025. R package version 0.1.0. Available at `https://github.com/AlexisDerumigny/PCBN`.

[7] J. Gauss and T. Nagler. Asymptotics for estimating a diverging number of parameters–with and without sparsity. *ArXiv preprint, arXiv:2411.17395*, 2024.

[8] I. H. Haff. Parameter estimation for pair-copula constructions. *Bernoulli*, 19(2):462–491, 2013.

[9] A. M. Hanea, D. Kurowicka, and R. Cooke. Hybrid method for quantifying and analyzing bayesian belief nets. *Quality and Reliability Engineering International*, 22:709–729, 2006.

[10] A. M. Hanea, D. Kurowicka, and R. Cooke. Mixed non-parametric continuous and discrete bayesian belief nets. In T. Bedford, J. Quigley, L. Walls, B. Alkali, A. Daneshkhah, and G. Hardman, editors, *Advances in Mathematical Modeling for Reliability*, chapter 1, pages 9–16. IOS Press, Amsterdam, 2008.

[11] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive computation and machine learning. MIT Press, 2009.

[12] D. Kurowicka and R. Cooke. Distribution-free continuous bayesian belief nets. *Modern Statistical and Mathematical Methods in Reliability*, 10:309–322, 2005.

[13] M. Maathuis, M. Drton, S. Lauritzen, and M. Wainwright. *Handbook of Graphical Models*. Chapman & Hall/CRC, USA, 1st edition, 2018.

[14] T. Nagler, U. Schepsmeier, J. Stoeber, E. C. Brechmann, B. Graeler, and T. Erhardt. *VineCopula: Statistical Inference of Vine Copulas*, 2021. R package version 2.4.3.

[15] M. Neil, D. Häger, and L. Andersen. Modelling operational risk in financial institutions using hybrid dynamic bayesian networks. *Journal of Operational Risk*, 4, 2009.

[16] O. Pourret, P. Naim, and B. Marcot. *Bayesian Networks. A Practical Guide to Applications*. John Wiley & Sons, 2008.

[17] M. Scutari, C. E. Graafland, and J. M. Gutiérrez. Who learns better bayesian network structures: Accuracy and speed of structure learning algorithms. *International Journal of Approximate Reasoning*, 115:235–253, 2019.

[18] H. Tsukahara. Semiparametric estimation in copula models. *Canadian Journal of Statistics*, 33(3):357–375, 2005.

A Results from [5]

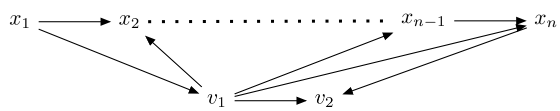
In this paper, we required some results concerning properties of minimal trails in restricted DAGs, Such results are presented and proved in the paper [5]. We summarize these results here to keep the paper self-contained.

Theorem A.1. *Let G be a DAG with no active cycles and let $v_1, v_2 \in V$ such that $v_1 \to v_2$. Suppose that*

$$v_1 \leftarrow x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons v_2 \tag{17}$$

is a shortest trail activated by the empty set starting with an arc $v_1 \leftarrow x_1$. Assume that $n \geq 1$. Then, for all $i \in \{1, \ldots, n\}$, $x_i \to x_{i+1}$ with the convention that $x_{n+1} := v_2$, and for all $i \in \{2, \ldots, n\}$, $v_1 \to x_i$.

This means that G contains the subgraph below.



Furthermore, the theorem also holds for shortest trails activated by the empty set and of the form

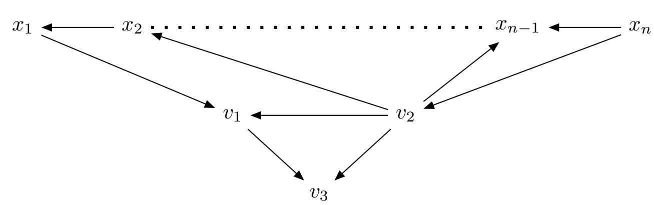
$$v_1 \leftarrow x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \to v_2 \tag{18}$$

with $n \geq 1$.

Theorem A.2. *Let G be a DAG with no active cycles and let $v_1, v_2, v_3 \in V$ such that $v_1, v_2 \in pa(v_3)$. Suppose that v_1 and v_2 are connected by a trail*

$$v_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \to v_2 \tag{19}$$

activated by the empty set with $\{x_i\}_{i=1}^n \cap pa(v_3) = \emptyset$ and $n \geq 1$. If this is a shortest such trail, then G contains the subgraph below, with the convention $x_0 := v_1$.



We will discuss trails between nodes, e.g. x_0 and x_{n+1}, for which all nodes on the trail are included in a certain subset $K \subseteq V$. In this case we say that the trail consists only of elements of K. This does not include the end-points (x_0 and x_{n+1}), i.e. these end-points may or may not be in K.

Definition A.1. *Let $G = (V, E)$ be a DAG, let $K \subseteq V$, and let $x_0 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_{n+1}$ be a trail. We say that the trail consists only of elements of K if $\forall i = 1, \ldots, n$, $x_i \in K$.*

We discuss the case when a shortest trail satisfying a certain property also satisfies a second property. Let us first formalize what is meant by a property of a trail.

Definition A.2 (Trail property)**.** *Let G be a DAG containing a trail $x_0 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_{n+1}$. A* ***property*** *$\mathfrak{P} := \mathfrak{P}(x_0, \ldots, x_{n+1})$ specifies the existence of certain arcs between the nodes on the trail. Here, we mean that $\mathfrak{P}$ states that E contains a certain set of arcs $\{x_i \to x_j;\, i \in I, j \in J\}$ with $I, J \subseteq \{0, 1, \ldots, n+1\}$.*

For instance, the following are regarded as trail properties:

- *The first arc of the trail points to the left; $x_0 \leftarrow x_1$.*
- *The i-th and j-th node on the trail are adjacent; $x_i \rightleftharpoons x_j$.*
- *The trail is of the form $x_0 \leftarrow x_1 \to x_2 \to \cdots \to x_{n-1} \to x_n$, and we have that $x_0 \to x_{n-1}$.*

Lemma A.1. *For a DAG G in $\mathcal{G}$, for a trail*

$$x_0 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_{n+1}, \tag{20}$$

let $\mathfrak{P}_1(x_0, x_1, \ldots, x_{n+1})$ and $\mathfrak{P}_2(x_0, x_1, \ldots, x_{n+1})$ be two properties. Let $\mathcal{G}$ be a set of DAGs such that

- *for any DAG $G = (V, E) \in \mathcal{G}$, for any $x_0, x_{n+1} \in V$, and for any shortest trail (20) between x_0 and x_{n+1} that satisfies $\mathfrak{P}_1$, the property $\mathfrak{P}_2$ holds.*
- *if G belongs to $\mathcal{G}$, then any graph obtained by removing vertices from G also belong to $\mathcal{G}$.*

Let $G = (V, E)$ be a DAG in $\mathcal{G}$, let $K \subseteq V$. Then for any shortest trail between x_0 and x_{n+1} that satisfies $\mathfrak{P}_1$ and that consists only of elements of K, the property $\mathfrak{P}_2$ still holds.

We start with a simple lemma about trails activated by the empty set.

Lemma A.2. *A trail is activated by the empty set if and only if it does not contain a converging connection.*

The lemma below states that if G contains a shortest trail $x_0 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons x_{n+1}$ activated by the empty set for which $x_0 \to v$ and $x_{n+1} \to v$ for some node $v \in V$, then for all $i = 1, \ldots, n$, $x_i \to v$.

Lemma A.3. *Let G be a DAG with no active cycles and let*

$$x_0 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons x_{n+1} \tag{21}$$

be a trail in G for some $n \geq 0$. If this is a shortest trail between x_0 and x_{n+1} activated by the empty set, then

(i) $ch(x_0) \cap ch(x_{n+1}) \subseteq \bigcap_{i=1}^{n} ch(x_i)$,

(ii) $\forall i = 1, \ldots, n,\ x_i \notin ch(x_0) \cap ch(x_{n+1})$.

In [5], the set of trails was defined as follows.

Definition A.3. *Let $X, Y, Z \subseteq V$ be disjoint subsets of V. We define $TRAILS\big(X, Y \,\big|\, Z\big)$ to be the set of trails from X to Y activated by Z.*

Moreover, the subtrails as well as the partial order $<_{TRAIL}$ on such sets are considered.

Definition A.4 (Subtrails)**.** *Let T be a trail in $TRAILS(X, Y \mid Z)$. Suppose that T takes the form*

$$x \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_{n_t(0)}^0 \to c_1 \leftarrow t_1^1 \rightleftharpoons \cdots \rightleftharpoons t_{n_t(1)}^1 \to c_2 \leftarrow \cdots\cdots\cdots \to c_C \leftarrow t_1^C \rightleftharpoons \cdots \rightleftharpoons t_{n_t(C)}^C \rightleftharpoons y.$$

*The following are referred to as the **subtrails** of T:*

$$\begin{aligned}
&x \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_{n_t(0)}^0 \to c_1,\\
&c_i \leftarrow t_1^i \rightleftharpoons \cdots \rightleftharpoons t_{n_t(i)}^i \to c_{i+1}, \ \textit{with } i \in \{1, \ldots, C-1\},\\
&c_C \leftarrow t_1^C \rightleftharpoons \cdots \rightleftharpoons t_{n_t(C)}^C \rightleftharpoons y.
\end{aligned}$$

The nodes on the subtrails are denoted by the symbol "t" where a superscript i indicates that t_j^i lies in between c_i and c_{i+1} with the conventions $c_0 := x$ and $c_{C+1} := y$. The subscript indicates its location on the subtrail. The length of a subtrail is formally denoted by $n_t(i)$, but we will often simply write $n := n_t(i)$.

Furthermore, we use the conventions $c_0 := x$, $c_{C+1} := y$, $t_0^i := c_i$ and $t_{n+1}^i := c_{i+1}$.

The 'minimal' according to $<_{TRAIL}$ trail in $TRAILS(X, Y \mid Z)$ has the following properties:

C1. It is a trail from X to Y activated by Z.

C2. It contains a smaller number of converging nodes not contained in Z.

C3. Under the restrictions above, it contains fewer converging connections.

C4. Under the restrictions above, the paths from converging nodes not contained in Z to its closest descendants are shorter.

C5. Under the restrictions above, it is a shorter such trail.

We also define the notion of a closest descendant in a trail.

Definition A.5 (Closest descendant)**.** *Let T be a trail in $TRAILS(X, Y \mid Z)$ and $i \in \{1, \ldots, C(T)\}$. If $c_i \notin Z$, then its **closest descendant** in Z is a node $Z(c_i) \in Z$ such that there exist a shortest path*

$$c_i \to d_1^i \to \cdots \to d_{n_Z(i)}^i \to Z(c_i)$$

with $d_j^i \notin Z$ for all $j = 1, \ldots, n_Z(i)$.

*Such a path is referred to as a **descendant path** of c_i. Its nodes on the descendant path are denoted by the symbol "d" where a superscript i indicates that d_j^i lies on the descendant path of c_i, and the subscript j indicates that it is the j-th node on this path. The length of the descendant path is formally denoted by $n_Z(i)$, but we will often simply write $n := n_Z(i)$. If $c_i \in Z$, we also say that $c_i = Z(c_i)$. Finally, we use the conventions $d_0^i := c_i$ and $d_{n+1}^i := Z(c_i)$.*

The following assumptions are often used below.

Assumption A.1. *Let $G = (V, E)$ be a DAG. The following conditions are assumed to be satisfied:*

1. G does not contain any active cycles, nor interfering v-structures.

2. $<$ is a well-ordering corresponding to G.

3. v is a node in V with $\mid pa(v) \mid > 0$.

4. All previous orders, i.e. $<_w$ with $w < v$, have already been determined by our algorithm.

5. O_v^k is a partial order determined by our algorithm with $k < \big|pa(v)\big|$.

Theorem A.3. *Let $X, Y, Z \subseteq V$ be three disjoint subsets. Assume that $TRAILS\big(X, Y \bigm| Z\big) \neq \emptyset$ and*

$$x \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_{n_t(0)}^0 \to c_1 \leftarrow\leftarrow \cdots\cdots\cdots \to c_C \leftarrow t_1^C \rightleftharpoons \cdots \rightleftharpoons t_{n_t(C)}^C \rightleftharpoons y. \tag{22}$$

be a minimal element of $TRAILS\big(X, Y \bigm| Z\big)$ with respect to the order $<_{TRAIL}$.

Then, the following properties hold:

(i) For all i, j, $t_j^i \notin X \sqcup Y \sqcup Z$ and $d_j^i \notin X \sqcup Y \sqcup Z$.

(ii) For all $i = 1, \ldots, C$, the trails $c_i \to d_1^i \to \cdots \to d_n^i \to Z(c_i)$ and $t_1^i \rightleftharpoons \cdots \rightleftharpoons t_n^i$ do not contain a chord. Furthermore, the trails $x \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0$ and $t_1^C \rightleftharpoons \cdots \rightleftharpoons t_n^C \rightleftharpoons y$ do not contain a chord.

(iii) If $c_i \to c_{i+1}$ and $c_{i+1} \in Z$, then $c_i \in Z$.

(iv) If $c_i \leftarrow c_{i+1}$ and $c_i \in Z$, then $c_{i+1} \in Z$.

(v) For all $i = 1, \ldots, C-1$, the i-th subtrail is a shortest trail between c_i and c_{i+1} starting with a leftward pointing arrow, ending with rightward pointing arrow, consisting of nodes in $V \setminus Z$ and with no converging connection. The C-th subtrail is a shortest trail between c_C and y starting with a leftward pointing arrow, consisting of nodes in $V \setminus Z$ and with no converging connection.

Definition A.6. *Let G be a DAG and K a subset of V. We say that K has* ***local relationships*** *if for all $v_1, v_2 \in K$ such that there exists a trail*

$$v_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons v_2$$

with $x_i \notin K$ for all $i = 1, \ldots, n$ and no converging connections, then v_1 and v_2 are adjacent.

Theorem A.4. *Let $X, Y, Z \subseteq V$ be three disjoint subsets and $Y \sqcup Z$ has local relationships (Definition A.6). Assume that $TRAILS\big(X, Y \bigm| Z\big) \neq \emptyset$ and let T a trail of the form (22) be a minimal element of $TRAILS\big(X, Y \bigm| Z\big)$ with respect to the order $<_{TRAIL}$. Then, the following properties hold.*

(i) The final converging node c_C is in Z.

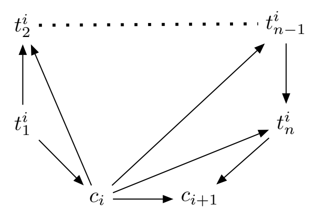
(ii) For all $i = 1, \ldots, C-1$, we have $c_i \in Z$ or $c_{i+1} \in Z$.

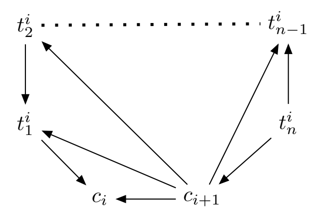
(iii) For all $i = 1, \ldots, C$, the nodes c_i and c_{i+1} are adjacent.

(iv) If this trail contains a total of $C > 0$ converging connections, then G contains the subgraph below.

$$\begin{tikzcd}
x \arrow[r, phantom, "\rightleftharpoons"] & t_1^0 \arrow[r, phantom, "\cdots\cdots"] & t_n^0 \arrow[r] & c_1 \arrow[r, phantom, "\rightleftharpoons"] \arrow[r, bend left, dash] & c_2 \arrow[r, phantom, "\cdots\cdots"] & c_{C-1} \arrow[r, phantom, "\rightleftharpoons"] \arrow[r, bend left, dash] & c_C \arrow[r, phantom, "\rightleftharpoons"] \arrow[r, bend left, dash] & y
\end{tikzcd}$$

Here, the curved lines represent one of the following two subgraphs.





Corollary A.1. *Let us consider the setting of Theorem A.4.*

(i) If the trail $c_1 \rightleftharpoons \cdots \rightleftharpoons c_C$ *takes the form* $c_1 \to \cdots \to c_C$*, then* $\forall i = 1, \ldots, C$, $c_i \in Z$.

(ii) If $c_1 \in Z$ *and the trail* $c_1 \rightleftharpoons \cdots \rightleftharpoons c_C$ *takes the form* $c_1 \leftarrow \cdots \leftarrow c_C$*, then* $\forall i = 1, \ldots, C$, $c_i \in Z$.

(iii) Let $i \in \{2, \ldots, C-1\}$*. If the trail* $c_{i-1} \rightleftharpoons c_i \rightleftharpoons c_{i+1}$ *is not a converging connection, then* $c_i \in Z$.

B Properties of B-sets and possible candidates

Informally, the lemma below states that two nodes in a B-set B_q are either d-separated given the empty set or any shortest trail activated by the empty set between them must be contained in B_q.

Lemma B.1. *Under Assumption A.1, let* $v \in V$*. Let* $q \in \{1, \ldots, Q(v)+1\}$ *and let* $w_1, w_2 \in B_q(v)$*. Then,* $\mathrm{dsep}\big(w_1, w_2 \,\big|\, \emptyset\big)$ *or any shortest trail activated by the empty set joining* w_1 *and* w_2 *must consist entirely of nodes contained in* B_q.

Proof. If $\mathrm{dsep}\big(w_1, w_2 \,\big|\, \emptyset\big)$, or if $w_1 = w_2$, then the proof of this lemma is completed. Therefore we can assume that they are not d-separated by the empty set and different from each other. Thus $\cancel{\mathrm{dsep}}\big(w_1, w_2 \,\big|\, \emptyset\big)$; let

$$w_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons w_2 \tag{23}$$

be a shortest trail between w_1 and w_2 activated by the empty set. First, we assume that $q \leq Q(v)$. Let $x_0 := w_1$, $x_{n+1} := w_2$, and b_q be a node corresponding to $B_q(v)$, see Definition 4.4. Because $w_1, w_2 \in B_q(v)$, we know that $v, b_q \in ch(w_1) \cap ch(w_2)$. By Lemma A.3 for all $i = 1, \ldots, n$, we have $v, b_q \in ch(x_i)$ and $x_i \in B_q(v) = pa(v) \cap pa(b_q)$.

If $q = Q(v)+1$, we are at the last stage of the algorithm and there is no b_q, but the same reasoning shows that for $i = 1, \ldots, n$, $x_i \in B_q(v) = pa(v)$. This concludes the proof. $\square$

A useful lemma proven in [5] (included without the proof in Lemma A.1) shows that one property of a trail that implies another will not only hold for shortest trails but also for shortest trails consisting of nodes in a subset $K \subseteq V$. However, this result cannot be directly applied to Lemma B.1, because the property that for all $i = 1, \ldots, n$, $x_i \in B_q(v)$ concerns a node v which is not on the trail. Therefore, we prove the generalization of Lemma A.1 in the corollary hereunder.

Corollary B.1. *Let G be a DAG with no active cycles nor interfering v-structures, and let $v \in V$. Let $q \in \{1, \dots, Q(v)+1\}$ and let $w_1, w_2 \in B_q(v)$. Let K be a set included in V. Then, w_1 and w_2 are either independent or for any shortest trail activated by the empty set joining w_1 and w_2 consisting of nodes in K must consist entirely of nodes contained in B_q.*

Proof. First, we assume that $q \leq B_q(v)$. Let b_q be a node corresponding to B_q. Let $G^* = (V^*, E^*)$ be the subgraph induced by the nodes in $V^* := \{v, w_1, w_2, b_q\} \cup K$. Note that v and b_q are children of both w_1 and w_2 in G^*. Therefore, by Lemma A.3(ii), any shortest trail between w_1 and w_2 in G^* activated by the empty set must not contain v nor b_q. This means that any shortest trail between w_1 and w_2 in G^* activated by the empty set must consist only of elements of K.

Consider a shortest trail in G

$$w_1 \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons w_2 \tag{24}$$

consisting of nodes in K, i.e. $\{x_i\}_{i=1}^n \subseteq K$. Therefore, it is a shortest trail activated by the empty set between w_1 and w_2 in G^*. We now apply Lemma B.1, since w_1 and w_2 belong to the B-set $B_q \cap V^*$ corresponding to v in the graph G^*. Therefore, for all $i = 1, \dots, n$, $x_i \in B_q$, completing the proof.

If $q = Q(v)+1$, then the proof is analogous to the previous case, but then with $V^* := (v, w_1, w_2) \cup K$. □

Informally, the lemma below states that if the set $PossCandInd(O_v^k)$ is empty, then there is a node in $B(O_v^k) \setminus O_v^k$ which is adjacent to a node in the set O_v^k.

Lemma B.2. *Under Assumption A.1, let $w \in B(O_v^k) \setminus O_v^k$ such that $\cancel{\mathrm{dsep}}(w, O_v^k \mid \emptyset)$, that is, $w \notin PossCandInd(O_v^k)$. Then, $ad(O_v^k) \cap B(O_v^k) \neq \emptyset$, where $ad(O_v^k)$ is the set of nodes adjacent to an element of O_v^k.*

Proof. By assumption, we have $\cancel{\mathrm{dsep}}(w, O_v^k \mid \emptyset)$. Therefore w must be connected to O_v^k by some trail activated by the empty set. We pick a shortest trail from w to O_v^k activated by the empty set, as

$$w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons o \tag{25}$$

where $o \in O_v^k$.

If $n = 0$, then $w \in B(O_v^k) \setminus O_v^k$ is adjacent to o and thus $w \in ad(O_v^k) \cap B(O_v^k)$.

Now, assume that $n > 0$. We will prove that $x_n \in ad(O_v^k) \cap B(O_v^k)$. Since we have a shortest trail between two nodes (w and o) in $B(O_v^k)$ with no chords, Lemma B.1 implies that all $x_i \in B(O_v^k)$. As a particular case, we have $x_n \in B(O_v^k)$.

If $x_n \in O_v^k$, then the trail $w \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n$ would be a shorter trail from w to O_v^k than the trail in (25). This is a contradiction, proving that $x_n \notin O_v^k$. Therefore $x_n \in B(O_v^k) \setminus O_v^k$. Note that x_n is adjacent to o, and thus $x_n \in ad(O_v^k) \cap B(O_v^k)$. This concludes the proof. □

By Proposition 4.1, we know that a node w is not a possible candidate for partial order O_v^k, if $\cancel{\mathrm{dsep}}(w, O_v^k \mid \emptyset)$ ($w \notin PossCandInd(O_v^k)$) and w and O_v^k are not adjacent ($w \notin PossCandIn(O_v^k) \sqcup PossCandOut(O_v^k)$). We will now prove an even stronger claim. That is, a node w is not a possible candidate to be added to a partial order O_v^k, if there exists an o in O_v^k such that:

- w and o are not adjacent.

- There exists a trail between w and o activated by the empty set which does not contain any nodes in O_v^k.

The lemma below provides a clear intuition into how the algorithm grows a partial order. For example, consider the a trail

$$o \rightleftharpoons w_1 \rightleftharpoons w_2 \rightleftharpoons \cdots \rightleftharpoons w_n,$$

with no converging connections where $o \in O_v^k$ and $\{w_i\}_{i=1}^n \subseteq pa(v) \setminus O_v^k$. In this case, we cannot add the node w_i to O_v^k for any $i \in \{2, \ldots, n\}$ since it is connected to o by an active trail $o \rightleftharpoons w_1 \rightleftharpoons \cdots \rightleftharpoons w_i$ consisting of nodes in $V \setminus O_v^k$. Consequently, we must add node w_1 before adding node w_i. If w_1 is added to O_v^k, then the same argument applies to the trail $w_1 \rightleftharpoons w_2 \rightleftharpoons \cdots \rightleftharpoons w_n$, i.e. we must add w_2 next. The recursion is clear; any node w_i can only be added after $w_1, \ldots, w_{i-1}$ have been added. So, the algorithm "walks" over trails with no converging connections, adding elements of these trails one node at a time, and it is only allowed to make "jumps" whenever a node is d-separated by the empty set from the current partial order.

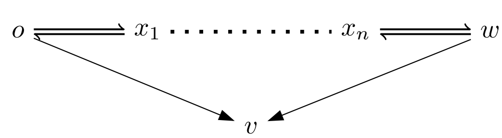
Lemma B.3. *Under Assumption A.1, let $o \in O_v^k$ and $w \in PossCand(O_v^k)$. If there exists a trail*

$$o \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons w, \tag{26}$$

with no converging connection such that $\forall i = 1, \ldots, n$, $x_i \in V \setminus O_v^k$, then w and o are adjacent.

Proof. We will employ an inductive argument, assuming that the lemma holds for all previous partial orders determined by the algorithm. By "previous partial orders" we mean all partial orders O_w^p with $w < v$ and $p \in \{1, \ldots, |pa(w)| - 1\}$, and O_v^p with $p \in \{1, \ldots, k-1\}$.

Without loss of generality we can assume that the trail (26) is a shortest trail between o and w with no converging connection and satisfying $\forall i = 1, \ldots, n$, $x_i \in V \setminus O_v^k$. Because o and w are parents of v by construction, G contains the subgraph below.



Because (26) has no converging connection, we know that $\cancel{\text{dsep}}\big(w, O_v^k \,\big|\, \emptyset\big)$. Thus, if $w \in PossCand(O_v^k)$, then we must have $w \in PossCandIn(O_v^k)$ or $w \in PossCandOut(O_v^k)$.

In the base case where $k = 1$ and $\left|O_v^k\right| = 1$, we know that $O_v^k = \{o\}$. Therefore we directly know that w and o are adjacent (because $w \in PossCandIn(O_v^k)$ or $w \in PossCandOut(O_v^k)$, so w must be connected to some node in O_v^k, and this must be o).

We now prove the induction step. If $n = 0$, then w and o are adjacent, which concludes the proof. We now assume $n > 0$. For this, we consider both cases depending on whether $w \in PossCandIn(O_v^k)$ or $w \in PossCandOut(O_v^k)$.

<u>Case 1: $w \in PossCandIn(O_v^k)$</u>. By definition of $PossCandIn(O_v^k)$ (Proposition 4.1), there exists an $\tilde{o} \in O_v^k$ such that $w \to \tilde{o}$ satisfying the following restrictions:

1. $pa(\tilde{o} \downarrow w) \subseteq O_v^k$.

2. $\text{dsep}\big(w, O_v^k \setminus (pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\}) \,\big|\, pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\}\big)$.

First, note that if $\tilde{o} = o$, then o and w are adjacent, completing the proof. Thus, we assume that $\tilde{o} \neq o$.

To satisfy the second restriction above, any trail between w and $O_v^k \setminus (pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\})$ must be blocked by $pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\}$. If we assume that $o \in O_v^k \setminus (pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\})$, then the trail (26) must be blocked by $pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\}$. Since this trail (26) contains no converging connections, there must be an $x_k \in (pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\})$ for some $k \in \{1, \ldots, n\}$. The first restriction combined with the definition of $\tilde{o}$ implies that $pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\} \subseteq O_v^k$, and thus $x_k \in O_v^k$ which contradicts the assumption of Lemma B.3 that x_i belongs to $V \setminus O_v^k$ for every $i = 1, \ldots, n$.

In the previous paragraph, we have proved that $o \notin O_v^k \setminus (pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\})$. Since $o \in O_v^k$, this implies that $o \in pa(\tilde{o} \downarrow w) \sqcup \{\tilde{o}\}$. Moreover, we assumed that $o \neq \tilde{o}$, and therefore $o \in pa(\tilde{o} \downarrow w)$ which means that $o \to \tilde{o}$ and $o <_{\tilde{o}} w$.

We have $\tilde{o} \in ch(o)$ and $\tilde{o} \in ch(w)$. Therefore, by Lemma A.3(i), $\tilde{o} \in ch(x_i)$ for all $i \in \{1, \ldots, n\}$. Therefore, all x_i must belong to $pa(\tilde{o}) = pa(\tilde{o} \downarrow w) \sqcup \{w\} \sqcup pa(\tilde{o} \uparrow w)$. None of them is equal to w since (26) is a shortest trail. By assumption, none of the x_i belong to O_v^k; the first restriction states that $pa(\tilde{o} \downarrow w) \subseteq O_v^k$; therefore all x_i belong to $pa(\tilde{o} \uparrow w)$. This means that $w <_{\tilde{o}} x_i$ for all $i = 1, \ldots, n$.

Now, we have $o <_{\tilde{o}} w <_{\tilde{o}} x_i$ for all $i \in \{1, \ldots, n\}$. This means that during the construction of $<_{\tilde{o}}$ in the algorithm we had $w \in PossCand(O_{\tilde{o}})$ for a partial order $O_{\tilde{o}}$ which contains o but not $\{x_i\}_{i=1}^n$. Therefore, by the induction hypothesis we obtain that w and o are adjacent, which finishes the proof for this case.

Case 2: $w \in PossCandOut(O_v^k)$. By Definition of $PossCandOut(O_v^k)$ (Proposition 4.1), there exists an $\tilde{o} \in O_v^k$ such that $\tilde{o} \to w$ satisfying the following conditions:

1. $pa(w \downarrow \tilde{o}) \subseteq O_v^k$.
2. $\text{dsep}\big(w, O_v^k \setminus (pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\}) \,\big|\, pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\}\big)$.

If $\tilde{o} = o$ the proof is complete. We now assume $\tilde{o} \neq o$.

If $o \notin pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$, then $o \in O_v^k \setminus (pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$ and therefore (26) is a trail from w to $O_v^k \setminus (pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$. By the second restriction above, this trail must be blocked by $pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$. Because this trail has no converging connection there must be an $i \in \{1, \ldots, n\}$ such that $x_i \in pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\} \subseteq O_v^k$ by the first restriction and the definition of $\tilde{o}$. This is a contradiction since by the assumption of the lemma x_i is in $V \setminus O_v^k$.

Therefore we have shown that $o \in pa(w \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$, which implies (by definition of this set) that o and w are adjacent, proving the lemma. □

Lemma B.3 immediately implies a very useful property of partial orders generated by our algorithm, which is proven in the corollary below.

Corollary B.2. *Under Assumption A.1, let $o_i, o_j \in O_v^k$, such that o_i (respectively o_j) is the i-th node (respectively j-th node) in the partial order O_v^k and $i \neq j$. If there exists a trail*

$$o_i \rightleftharpoons x_1 \rightleftharpoons \cdots \rightleftharpoons x_n \rightleftharpoons o_j, \tag{27}$$

with no converging connection such that $\forall m = 1, \ldots, n$, $x_m \in V \setminus O_v^k$, then o_i and o_j are adjacent.

Proof. Without loss of generality, we can assume that $i < j$. This means that $o_j \in PossCand(O_v^{j-1})$, with $o_i \in O_v^{j-1}$. Remark that $O_v^{j-1} \subseteq O_v^k$. Therefore, for all $m = 1, \ldots, n$, $x_m \in V \setminus O_v^{j-1}$. Hence, by Lemma B.3, o_i and o_j are adjacent. $\square$

We now prove a lemma which states that sets which are d-separated cannot be adjacent. It is quite trivial but it will be useful in Lemma B.5.

Lemma B.4. *Let $G = (V, E)$ be a DAG and X, Y, Z subsets of V such that* $\operatorname{dsep}\big(X, Y \,\big|\, Z\big)$. *Then X and Y cannot be adjacent, in the sense that $\forall x, y \in X \times Y$, $x \not\rightleftharpoons y$.*

Proof. Let $x \in X$ and $y \in V$ such that $x \rightleftharpoons y$. y is adjacent to $x \in X$ so the trail $x \rightleftharpoons y$ is active given Z. This shows that $\cancel{\operatorname{dsep}}\big(X, \{y\} \,\big|\, Z\big)$, which contradicts $\operatorname{dsep}\big(X, Y \,\big|\, Z\big)$. $\square$

The lemma below states that under certain conditions an arc between a node $w \in B(O_v^k) \setminus O_v^k$ and a node $o_i \in O_v^k$ implies the existence of another node $\tilde{o} \in O_v^k$ such that $w \to \tilde{o} \in E$ and $o_i \to \tilde{o} \in E$.

Lemma B.5. *Following Definition 4.1, let us write the partial order O_v^k as $O_v^k = (o_1, \ldots, o_k)$. Under Assumption A.1, let $i \in \{1, \ldots, k\}$ and $w \in B(O_v^k) \setminus O_v^k$. If $w \to o_i$ and $pa(o_i \uparrow w) \cap O_v^k \neq \emptyset$, then there exists an $\tilde{o} \in O_v^k$ such that $\tilde{o} \neq o_i$ and G contains the subgraph below.*

$$\begin{tikzcd} w \arrow[d] \arrow[dr] & \\ o_i \arrow[r] & \tilde{o} \end{tikzcd}$$

Proof. Note that $w \in pa(o_i) = pa(o_i \downarrow w) \sqcup \{w\} \sqcup pa(o_i \uparrow w)$. Since $pa(o_i \uparrow w) \cap O_v^k \neq \emptyset$, let o_j be its maximum element according to $<_{o_i}$. Consequently, $o_j \to o_i$ and $w <_{o_i} o_j$.

Assume that there exists an $\tilde{o} \in O_v^k$ such that G contains the v-structure $o_i \to \tilde{o} \leftarrow o_j$. Thus, G contains the subgraph below.

$$\begin{tikzcd} w \arrow[d] & o_j \arrow[dl] \arrow[d] \\ o_i \arrow[r] & \tilde{o} \end{tikzcd}$$

Observe that $o_j \in pa(o_i)$ and $o_j \in pa(\tilde{o})$; by definition of the B-sets, $o_j \in B(o_i, \tilde{o}) = pa(o_i) \cap pa(\tilde{o})$. By Corollary 4.2, since $w <_{o_i} o_j$, we obtain $w \to \tilde{o}$, giving us the desired subgraph and finishing the proof under the assumption of existence of $\tilde{o}$.

There remains to prove the existence of such an $\tilde{o}$. We consider two cases; when $i < j$ and $j < i$.

Case 1: $i < j$. In this case, since $o_j \to v$, o_j was added at the step $j - 1$, and therefore we must have $o_j \in PossCand(O_v^{j-1})$. Because $i < j$, we obtain $o_i \in O_v^{j-1}$. Therefore, $\cancel{\operatorname{dsep}}\big(o_j, O_v^{j-1} \,\big|\, \emptyset\big)$ because of the arc $o_j \to o_i$. Hence, $o_j \notin PossCandInd(O_v^{j-1})$, and therefore o_j must be in $PossCandIn(O_v^{j-1})$ or $PossCandOut(O_v^{j-1})$. We consider both cases.

- $\underline{o_j \in PossCandIn(O_v^{j-1})}$: By Proposition 4.1 there must be a node $\tilde{o} \in O_v^{j-1}$ such that $o_j \to \tilde{o}$ satisfying following restrictions:

  1. $pa(\tilde{o} \downarrow o_j) \subseteq O_v^{j-1}$.
  2. $\mathrm{dsep}\big(o_j, O_v^{j-1} \setminus (pa(\tilde{o} \downarrow o_j) \sqcup \{\tilde{o}\}) \,\big|\, pa(\tilde{o} \downarrow o_j) \sqcup \{\tilde{o}\}\big)$.

  Remark that $w \in pa(o_i \downarrow o_j)$ and $w \notin O_v^k \supseteq O_v^{j-1}$, and thus $pa(o_i \downarrow o_j) \nsubseteq O_v^{j-1}$. This shows that $\tilde{o} \neq o_i$ otherwise the first restriction could not be satisfied.

  By combining Lemma B.4 and the second restriction, no node in $O_v^{j-1} \setminus (pa(\tilde{o} \downarrow o_j) \sqcup \{\tilde{o}\})$ can be adjacent to o_j. Because we have the arc $o_j \to o_i$ we can deduce that $o_i \notin O_v^{j-1} \setminus (pa(\tilde{o} \downarrow o_j) \sqcup \{\tilde{o}\})$.

  Since $o_i \in O_v^{j-1}$, this means that $o_i \in pa(\tilde{o} \downarrow o_j) \sqcup \{\tilde{o}\}$, and therefore $o_i \to \tilde{o}$. Now, we have $o_j \to \tilde{o}$ and $o_i \to \tilde{o}$ which is the desired v-structure.

- $\underline{o_j \in PossCandOut(O_v^{j-1})}$: By Proposition 4.1 there must be a node $\tilde{o} \in O_v^{j-1}$ such that $\tilde{o} \to o_j$ and the following restrictions are satisfied:

  1. $pa(o_j \downarrow \tilde{o}) \subseteq O_v^{j-1}$.
  2. $\mathrm{dsep}\big(o_j, O_v^{j-1} \setminus (pa(o_j \downarrow \tilde{o}) \sqcup \{\tilde{o}\}) \,\big|\, pa(o_j \downarrow \tilde{o}) \sqcup \{\tilde{o}\}\big)$.

  If $\tilde{o} = o_i$, then G contains the cycle $o_i \to o_j \to o_i$, which is a contradiction, and thus $\tilde{o} \neq o_i$.

  Combining Lemma B.4 and the second restriction, no point in $O_v^{j-1} \setminus (pa(o_j \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$ can be adjacent to o_j. Because we have the arc $o_j \to o_i$ we can deduce that $o_i \notin O_v^{j-1} \setminus (pa(o_j \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$.

  Since $o_i \in O_v^{j-1}$, this means that $o_i \in pa(o_j \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$, and therefore $o_i \to o_j$. This provides the cycle $o_i \to o_j \to o_i$ which gives a contradiction, showing that o_j cannot be in $PossCandOut(O_v^{j-1})$.

Case 2: $j < i$. In this case, since $o_i \to v$, o_i was added at the step $i-1$, therefore we must have $o_i \in PossCand(O_v^{i-1})$. Because $j < i$, we obtain $o_j \in O_v^{i-1}$. Hence, $\cancel{\mathrm{dsep}}\big(o_i, O_v^{i-1} \,\big|\, \emptyset\big)$ due to existence of the arc $o_j \to o_i$. We get $o_i \notin PossCandInd(O_v^{i-1})$. Thus, o_i must be in $PossCandIn(O_v^{i-1})$ or $PossCandOut(O_v^{i-1})$. Both cases are considered below.

- $\underline{o_i \in PossCandIn(O_v^{i-1})}$: By Proposition 4.1 there must be a node $\tilde{o} \in O_v^{i-1}$ such that $o_i \to \tilde{o}$ satisfying:

  1. $pa(\tilde{o} \downarrow o_i) \subseteq O_v^{i-1}$.
  2. $\mathrm{dsep}\big(o_i, O_v^{i-1} \setminus (pa(\tilde{o} \downarrow o_i) \sqcup \{\tilde{o}\}) \,\big|\, pa(\tilde{o} \downarrow o_i) \sqcup \{\tilde{o}\}\big)$.

  Note that $\tilde{o} = o_j$ creates the cycle $o_j \to o_i \to o_j$ which is a contradiction, and thus $\tilde{o} \neq o_j$.

  As before, combining Lemma B.4, the second restriction, and the arc $o_j \to o_i$ implies that $o_j \notin O_v^{i-1} \setminus (pa(\tilde{o} \downarrow o_i) \sqcup \{\tilde{o}\})$, and therefore $o_j \in pa(\tilde{o} \downarrow o_i) \sqcup \{\tilde{o}\}$ which means that $o_j \to \tilde{o}$, giving us the desired v-structure.

- $\underline{o_i \in PossCandOut(O_v^{i-1})}$: By Proposition 4.1 there must be a node $\tilde{o} \in O_v^{i-1}$ such that $\tilde{o} \to o_i$ satisfying:

1. $pa(o_i \downarrow \tilde{o}) \subseteq O_v^{i-1}$.
2. $\text{dsep}\big(o_i, O_v^{i-1} \setminus (pa(o_i \downarrow \tilde{o}) \sqcup \{\tilde{o}\}) \,\big|\, pa(o_i \downarrow \tilde{o}) \sqcup \{\tilde{o}\}\big)$.

Remark that $w \in pa(o_i \downarrow o_j)$ and $w \notin O_v^k \supseteq O_v^{j-1}$, and thus $pa(o_i \downarrow o_j) \not\subseteq O_v^{j-1}$. This shows that $\tilde{o} \neq o_j$ otherwise the first restriction could not be satisfied.

Combining Lemma B.4, the second restriction, and the existence of the arc $o_j \to o_i$ implies that $o_j \notin O_v^{i-1} \setminus (pa(o_i \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$. Since $o_j \in O_v^{i-1}$ this means that $o_j \in pa(o_i \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$, and thus $o_j <_{o_i} \tilde{o}$. Therefore, $\tilde{o} \in pa(o_i \uparrow w)$ since $w <_{o_i} o_j$. However, this is a contradiction since $o_j \neq \tilde{o}$ was chosen to be the maximum element in $pa(o_i \uparrow w) \cap O_v^k$. This shows that it cannot happen that o_i is in $PossCandOut(O_v^{i-1})$.

□

C Lemmas to construct sequences of nodes

C.1 Possible candidates by incoming arc

Assumption C.1. *Assumption A.1 holds. Furthermore, $w \in B(O_v^k) \setminus O_v^k$ and $o \in O_v^k$ are nodes such that (i) $w \to o \in E$ (ii) $pa(o \downarrow w) \subseteq O_v^k$, (iii) $pa(o \uparrow w) \cap O_v^k = \emptyset$.*

Lemma C.1. *Assume that Assumption C.1 holds and that $w \notin PossCandIn(O_v^k)$. Then there exists a trail from w to an element of $O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$ which is activated by $pa(o \downarrow w) \sqcup \{o\}$ and contains no converging connections.*

Proof. Since $w \notin PossCandIn(O_v^k)$, w cannot be a possible candidate by the incoming arc $w \to o$. This means that one of the two restrictions must be violated:

1. $pa(o \downarrow w) \subseteq O_v^k$.
2. $\text{dsep}\big(w, O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big) \,\big|\, pa(o \downarrow w) \sqcup \{o\}\big)$.

The first restrictions is satisfied by the assumption of the lemma. Therefore, the second restrictions must be violated, meaning that $\cancel{\text{dsep}}\big(w, O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big) \,\big|\, pa(o \downarrow w) \sqcup \{o\}\big)$. Hence, the set $TRAILS\big(X, Y \,\big|\, Z\big)$ is not empty. Consequently, there exists a minimal trail

$$w \rightleftharpoons \cdots \to c_1 \leftarrow \cdots \to c_2 \leftarrow \cdots\cdots\cdots \to c_C \leftarrow \cdots \rightleftharpoons y$$

in $TRAILS\big(X, Y \,\big|\, Z\big)$ according to $<_{TRAIL}$ with $X = \{w\}$, $Y := O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$ and $Z := pa(o \downarrow w) \sqcup \{o\}$.

Therefore, $TRAILS\big(X, Y \,\big|\, Z\big)$ is not empty. Lemma C.2(vii) implies that the existence of a minimal trail in $TRAILS\big(X, Y \,\big|\, Z\big)$ containing no converging connections. This concludes the proof of Lemma C.1.

□

Lemma C.2. *Assume that Assumption C.1 holds and take a minimal trail*

$$w \rightleftharpoons \cdots \to c_1 \leftarrow \cdots \to c_2 \leftarrow \cdots\cdots\cdots \to c_C \leftarrow \cdots \rightleftharpoons y \tag{28}$$

in the set $TRAILS\big(X, Y \,\big|\, Z\big)$ *according to* $<_{TRAIL}$ *with* $X := \{w\}$, $Y := O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$ *and* $Z := pa(o \downarrow w) \sqcup \{o\}$.

Then, the following statements hold:

(i) If the node o *is included in the trail, then it must be the first node, i.e.* $o = c_1$. *If this is the case, then* $w \to o$ *is the first subtrail.*

(ii) If $c_1 \in Z = pa(o \downarrow w) \sqcup \{o\}$ *then* G *contains one of the three subgraphs below presented in Figure 12.*

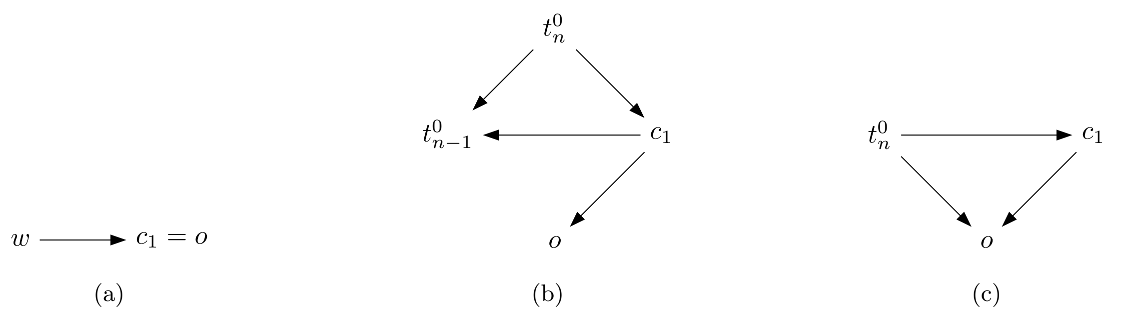


Figure 12: Subgraphs for which one must be included in G in case that $c_1 \in Z$.

(iii) If $C > 1$, *then without loss of generality we can assume that* $c_{C-1} \leftarrow c_C$, *in the sense that there exists a minimal trail (according to* $<_{TRAIL}$*) in* $TRAILS\big(X, Y \,\big|\, Z\big)$ *such that* $c_{C-1} \leftarrow c_C$.

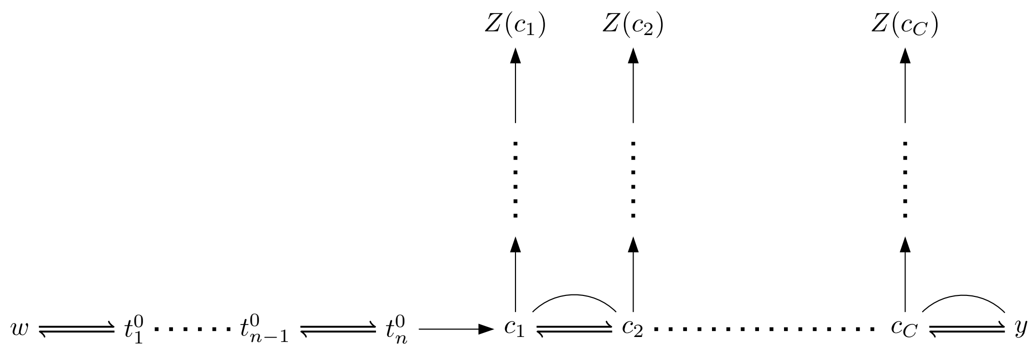
(iv) The node y *is not in* $pa(o)$.

(v) There exists a node $\tilde{y} \in Y = O_v^k \setminus \big(pa(o) \sqcup \{o\}\big)$ *such that* $o \to \tilde{y}$ *and* $\forall i = 1, \ldots, C$, $Z(c_i) \to \tilde{y}$ *(whenever* $Z(c_i) \neq \tilde{y}$*). Furthermore, if* $\tilde{y} \neq y$, *then* $y \to \tilde{y}$.

(vi) The total number of converging nodes C *cannot be strictly larger than* 1.

(vii) The trail has no converging connections, i.e. $C = 0$.

Proof of Lemma C.2. First, note that $Y \sqcup Z = O_v^k$ have local relationships in the sense of Definition A.6, by Corollary B.2. Therefore, by Theorem A.4(iv), G contains the subgraph below with $y \in Y = O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$.



Each of these properties is proved, respectively in the following Sections C.1.1, C.1.2, C.1.3, C.1.4, C.1.5, C.1.6 and C.1.7.

C.1.1 Proof of Lemma C.2(i)

Naturally, if the trail (28) contains the node $o \in Z = pa(o \downarrow w) \sqcup \{o\}$, then it must correspond to a converging connection. Otherwise, (28) would be blocked by Z.

Consider the case when a node c_i with $i \in \{2, \ldots, C\}$ is equal to o. Then, the trail

$$w \to o \leftarrow \cdots \to c_{i+1} \leftarrow \cdots\cdots\cdots \to c_C \leftarrow \cdots \rightleftharpoons y$$

would be a better trail than (28) which is a contradiction. Hence, if o is located along the trail it must be equal to c_1.

In this case the subtrail $w \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to o$ takes the form $w \to o$, since we picked a minimal trail. This concludes the proof of (i).

C.1.2 Proof of Lemma C.2(ii)

Consider the subtrail

$$w \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to c_1. \tag{29}$$

If o is in the trail (28), then by (i), it is the first node along the trail, i.e. G contains the subgraph in Figure 12a.

Suppose that o is not located along the trail. Therefore, $c_1 \neq o$. Because we assumed that $c_1 \in Z = pa(o \downarrow w) \sqcup \{o\}$, we know that c_1 belongs to $pa(o \downarrow w)$ and therefore $c_1 \to o$. If t_n^0 is in $pa(o)$, then $t_n^0 \to o$. In this case, we find the subgraph in Figure 12c.

We will now show that if t_n^0 is not in $pa(o)$ and o is not in (28), then G must contain the subgraph in Figure 12b. First, we define an integer

$$p := \max\{i \in \{1, \ldots, n\};\ t_i^0 \in pa(o)\}$$

such that t_p^0 is the furthest node from w in trail (29) contained in the set $pa(o)$. By Theorem A.3(ii), the subtrail

$$t_p^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to c_1$$

contains no chords. Moreover, the nodes t_p^0 and c_1 are in $pa(o)$, and the nodes $t_{p+1}^0, \ldots, t_n^0$ are not in $pa(o)$.

Remark that $t_p^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to c_1$ is a shortest trail activated by the empty set from t_p^0 to c_1 ending with a rightward pointing arrow consisting of nodes in $V \setminus Z$. Therefore, we may apply Lemma A.1 and Theorem A.2 (with $v_1 = t_p^0$, $v_2 = c_1$ and $v_3 = o$ in the notation of Theorem A.2) to find that $t_n^0 \to t_{n-1}^0$ and $c_1 \to t_{n-1}^0$. Since we also know that $c_1 \to o$, we conclude that to find that G must contain the subgraph in Figure 12b.

C.1.3 Proof of Lemma C.2(iii)

By Theorem A.4(iii), we know that c_{C-1} and c_C are adjacent. If $c_{C-1} \to c_C$ then the proof is completed. There only remains to study the case where $c_{C-1} \to c_C$. From Theorem A.4(iii), it follows that for all $i = 1, \ldots, C$, the nodes c_i and c_{i+1} are adjacent. By Lemma C.5, we know that for all $i = 2, \ldots, C$, the trail $c_{i-1} \leftarrow c_i \to c_{i+1}$ cannot be present. Therefore, we get that for all $i = 1, \ldots, C$, $c_i \to c_{i+1}$. From Theorem A.4(i), it follows that $c_C \in Z$. Combining this with Corollary A.1(i), we find that $\forall i = 1, \ldots, C-1$, $c_i \in Z$. In particular we have that $c_1 \to c_2$ and $c_1, c_2 \in Z$.

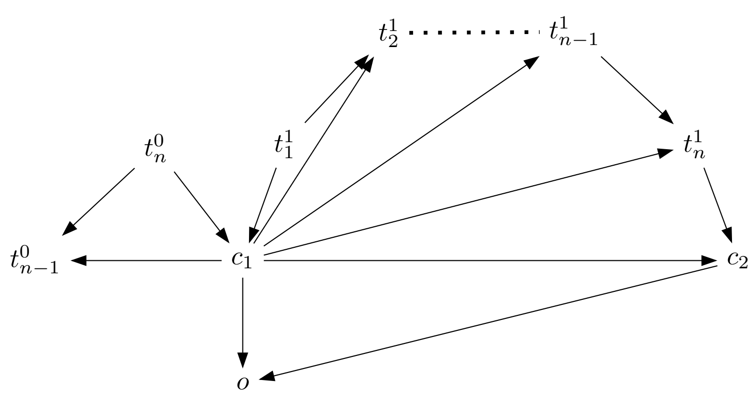
We will show that the arc $c_1 \to c_2$ with $c_1, c_2 \in Z$ leads the existence of another minimal trail, that satisfies $c_{C-1} \leftarrow c_C$. Since $c_1 \in Z$ we can apply (ii), to find that G contains one of three subgraphs in Figure 12. We consider each case separately.

Case 1: Subgraph 12a.

In this case $c_1 = o$. Since $c_1 \to c_2$, we have that $o \to c_2$. Moreover, because $c_2 \in Z = pa(o \downarrow w) \sqcup \{o\}$, we know that $c_2 \to o$, that leads to the cycle $c_2 \to o \to c_2$ which is a contradiction.

Case 2: Subgraph 12b.

Note that o is not equal to c_1 (otherwise we would be in the previous case) and therefore $c_2 \neq o$. Since $c_2 \in Z = pa(o \downarrow w) \sqcup \{o\}$, we get that $c_2 \to o$. Combining this and Figure 12b with Theorem A.4(iv), we obtain that G contains the subgraph below.



Here, we have

$$t_1^1 \in B(c_1, t_2^1),$$
$$t_n^0 \in B(c_1, t_{n-1}^0).$$

Since G does not contain any interfering v-structures, this means that $t^0_n \to t^1_2$ or $t^1_1 \to t^0_{n-1}$. These arcs provide us with the following respective trails

$$w \rightleftharpoons \cdots \rightleftharpoons t^0_{n-1} \rightleftharpoons t^0_n \to t^1_2 \rightleftharpoons \cdots\cdots\cdots \rightleftharpoons y,$$
$$w \rightleftharpoons \cdots \rightleftharpoons t^0_{n-1} \leftarrow t^1_1 \rightleftharpoons t^1_2 \rightleftharpoons \cdots\cdots\cdots \rightleftharpoons y,$$

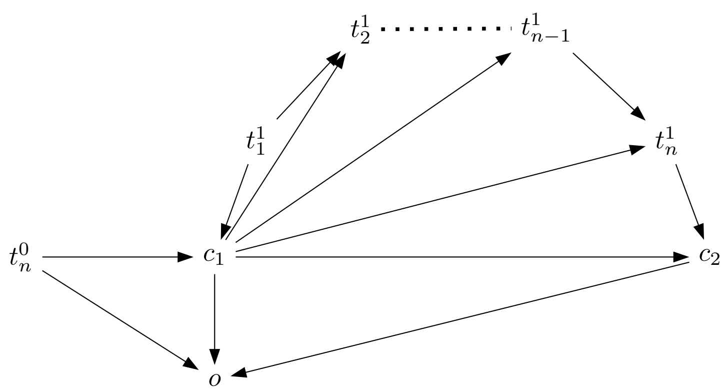
which are both better than (28) in the sense of $<_{TRAIL}$. Indeed, (28) can be rewritten as

$$w \rightleftharpoons \cdots \rightleftharpoons t^0_{n-1} \rightleftharpoons t^0_n \to c_1 \leftarrow t^1_1 \rightleftharpoons t^1_2 \rightleftharpoons \cdots\cdots\cdots \rightleftharpoons y.$$

We therefore get a contradiction.

<u>Case 3: Subgraph 12c.</u>

In this case, by Theorem A.4(iv), G contains the subgraph below, where $c_2 \to o$ because $c_2 \in Z = pa(o \downarrow w) \sqcup \{o\}$ and $c_2 \neq o$.



Here, we have $t^0_n \in B(c_1, o)$ and $t^1_1 \in B(c_1, t^1_2)$. By the same argument as above, this means that $t^0_n \to t^1_2$ or $t^1_1 \to o$. The former arc results in a trail from w to y, which is a better trail than (28), and therefore a contradiction. Hence, we must have the arc $t^1_1 \to o$. This arc provides us with the trail

$$w \to o \leftarrow t^1_1 \rightleftharpoons \cdots\cdots\cdots \rightleftharpoons y \tag{30}$$

which is better than the trail (28), that is

$$w \rightleftharpoons \cdots \rightleftharpoons t^0_{n_t(0)} \to c_1 \leftarrow t^1_1 \rightleftharpoons \cdots\cdots\cdots \rightleftharpoons y,$$

unless $n_t(0) = 0$. In this case, (30) is also a minimal trail in $TRAILS\big(X, Y \,\big|\, Z\big)$.

Note that the converging connections of the trail (30) are $o, c_2, \ldots, c_C$. Combining the fact that $o \leftarrow c_2$ with Lemma C.5 we must have

$$o \leftarrow c_2 \leftarrow c_3 \leftarrow \cdots \leftarrow c_C.$$

So, we have proven that there exists a minimal trail in $TRAILS\big(X, Y \,\big|\, Z\big)$ containing an arc $c_{C-1} \leftarrow c_C$, completing the proof of (iii).

C.1.4 Proof of Lemma C.2(iv)

By definition, $y \in Y = O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$. Observe that

$$\begin{aligned} Y \cap pa(o) &= \big(O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)\big) \cap \big(pa(o \downarrow w) \sqcup \{o\} \sqcup pa(o \uparrow w)\big) \\ &= \big(O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)\big) \cap pa(o \uparrow w) \\ &\subseteq O_v^k \cap pa(o \uparrow w) = \emptyset, \end{aligned}$$

by Assumption C.1, hence $y \notin pa(o)$.

C.1.5 Proof of Lemma C.2(v)

Suppose that o is located on (28). By (i), this means that $o = c_1$, and therefore G contains the trail

$$o \leftarrow \cdots \rightarrow c_2 \leftarrow \cdots\cdots\cdots \rightarrow c_C \leftarrow \cdots \rightleftharpoons y.$$

that satisfies all conditions for Lemma C.6 (by applying Theorem A.3(i)). Remember that $w \rightarrow o$ by Assumption C.1. If $o \neq c_1$, then the trail

$$o \leftarrow w \rightleftharpoons \cdots \rightarrow c_1 \leftarrow \cdots \rightarrow c_2 \leftarrow \cdots\cdots\cdots \rightarrow c_C \leftarrow \cdots \rightleftharpoons y$$

also satisfies the conditions of Lemma C.6 (by applying Theorem A.3(i)).

Therefore, in both cases we can apply Lemma C.6 to find that there exists an $\tilde{y}$ in O_v^k such that $o \rightarrow \tilde{y}$, $y \rightarrow \tilde{y}$ and for all $i = 1, \ldots, C$, $Z(c_i) \rightarrow \tilde{y}$ (whenever this does not create a self-loop $\tilde{y} \rightarrow \tilde{y}$, i.e. in the case where $\tilde{y}$ would be equal to o, y or some $Z(c_i)$ for $i \in \{1, \ldots, C\}$). It remains to show that this node $\tilde{y}$ is in $Y = O_v^k \setminus \big(pa(o) \sqcup \{o\}\big)$, Therefore, we only have to show that $\tilde{y}$ cannot be in $pa(o) \sqcup \{o\}$.

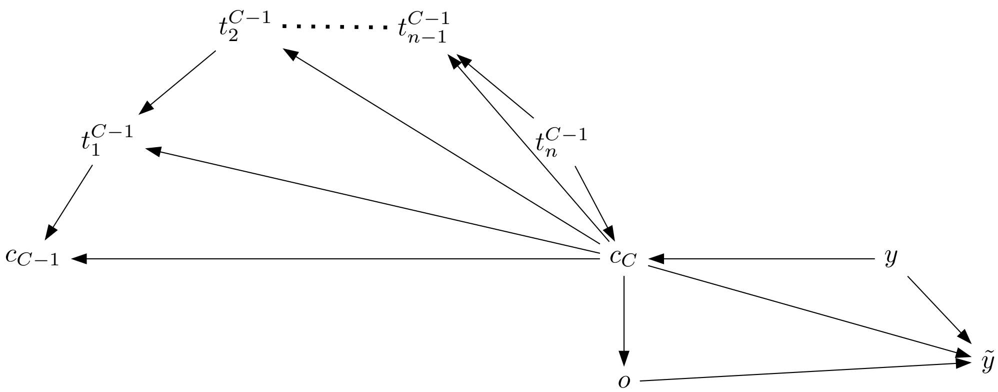
If $\tilde{y} = o$, then G contains the arc $y \rightarrow o$. This contradicts (iv) which states that $y \notin pa(o)$. If $\tilde{y} \in pa(o)$ then $\tilde{y} \rightarrow o$, and hence G contains the cycle $\tilde{y} \rightarrow o \rightarrow \tilde{y}$ which is a contradiction (because we showed above that $o \rightarrow \tilde{y}$).

C.1.6 Proof of Lemma C.2(vi)

Assume that (28) has $C > 1$ converging connection. The end of the trail can have several different types of structures. By (iii) we can assume that $c_{C-1} \leftarrow c_C$ without loss of generality. If $c_C \rightarrow y$, we obtain the subgraph $c_{C-1} \leftarrow c_C \rightarrow c_{C+1}$, which is a contradiction by Lemma C.5. Therefore $c_C \leftarrow y$.

Remark that by Theorem A.4(i) the node c_C is in the set $Z := pa(o \downarrow w) \sqcup \{o\}$. Note that, by (i), $c_C \neq o$, and therefore $c_C \in pa(o \downarrow w)$, so $c_C \rightarrow o$. Consequently, the graph contains the trail $y \rightarrow c_C \rightarrow o$. This means that the node $\tilde{y} \in Y$ from (v) cannot be equal to y. Indeed, this would lead to the cycle $y \rightarrow c_C \rightarrow o \rightarrow y$ which is a contradiction.

Furthermore, $c_C \in Z$, therefore $Z(c_C) = c_C \rightarrow \tilde{y}$ by (v). Combining the previous results with Theorem A.4(iv) gives the subgraph below.



Here, we have $t_n^{C-1} \in B(c_C, t_{n-1}^{C-1})$ and $y \in B(c_C, \tilde{y})$. Since G does not contain interfering v-structures, we must have $t_n^{C-1} \to \tilde{y}$ or $y \to t_{n-1}^{C-1}$. Both arcs provide a trail from w to a node in $Y := O_v^k \setminus \big(pa(o \downarrow w) \sqcup \{o\}\big)$ which is a better trail than (28). Indeed, the trails

$$w \rightleftharpoons \cdots\cdots\cdots \leftarrow t_n^{C-1} \to \tilde{y},$$
$$w \rightleftharpoons \cdots\cdots\cdots \leftarrow t_{n-1}^{C-1} \leftarrow y,$$

contain one fewer converging connection than (28), which is

$$w \rightleftharpoons \cdots\cdots\cdots \leftarrow t_n^{C-1} \to c_C \leftarrow y.$$

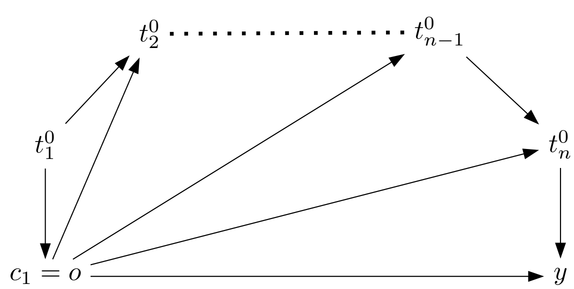
This leads to a contradiction because (28) was assumed to be a minimal trail and the proof of (vi) is concluded.

C.1.7 Proof of Lemma C.2(vii)

By (vi), the trail (28) has either 0 or 1 converging connection. If it has zero converging connection, then the existence of this trail completes the proof of (vii). Therefore we assume that (28) has exactly one converging connection, i.e. $C = 1$. Furthermore, by Theorem A.4(i) we know that $c_C = c_1 \in Z := pa(o \downarrow w) \sqcup \{o\}$. This means that we can apply (ii) to find that G contains one of the three subgraphs in Figure 12. We consider each subgraph separately. Furthermore, for each case we will consider two sub-cases; when $c_1 \to y$ and when $c_1 \leftarrow y$, since c_1 and y are adjacent by Theorem A.4(iii).

Case 1: Subgraph 12a.

In this case the node c_1 is equal to o. Since $y \notin pa(o)$ by (iv), the arc $c_1 \leftarrow y$ cannot be present. Therefore, we must have $c_1 \to y$. Thus, by Theorem A.4(iv) we know that G contains the subgraph below.



The node t_1^0 is in $V \setminus O_v^k$ by Theorem A.3(i), and it is also in $pa(o)$. This means that $t_1^0 \in pa(o) \setminus O_v^k$. By Assumption C.1, we have $pa(o \downarrow w) \subseteq O_v^k$ and $o \in O_v^k$. Therefore, $pa(o) \setminus O_v^k = \big(pa(o \downarrow w) \sqcup \{w\} \sqcup pa(o \uparrow w)\big) \setminus O_v^k \subseteq pa(o \uparrow w) \sqcup \{w\}$. Thus, $t_1^0 \in pa(o \uparrow w) \sqcup \{w\}$. If $t_1^0 = w$, we find a shorter trail than (28), which is a contradiction. Therefore $t_1^0 \in pa(o \uparrow w)$ and so $w <_O t_1^0$.

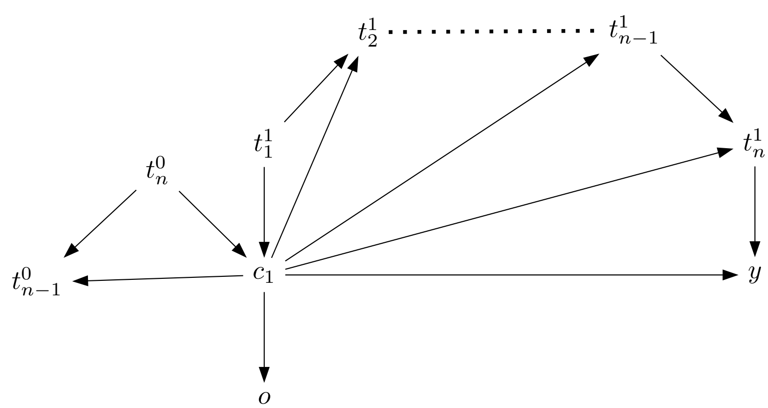
Because the parental order $<_O$ has been determined by our algorithm, it abides by the B-sets, see Corollary 4.1. Therefore, any B-set corresponding to the node o which contains t_1^0 must also contain w. Remark that $t_1^0 \in B(o, t_2^0)$. Consequently, we have that $w \in B(o, t_2^0)$. This means that $w \to t_2^0$ leads to the trail

$$w \to t_2^0 \to \cdots \to t_n^0 \to y$$

which contains no converging connections. Thus, this trail is better than the trail (28) which is a contradiction.

Case 2: Subgraph 12b.

Note that both cases $c_1 \to y$ and when $y \to c_1$ must be considered. For both cases we have that $c_1 \in Z = pa(o \downarrow w) \sqcup \{o\}$, and therefore $c_1 \to o$. First, let us assume that $c_1 \to y$, then by Theorem A.4(iv) we know that G contains the subgraph below.



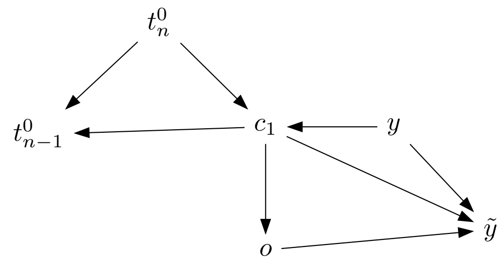
Here, we have that $t_n^0 \in B(c_1, t_{n-1}^0)$ and $t_1^1 \in B(c_1, t_2^1)$. Since G does not contain any interfering v-structures, we must have $t_n^0 \to t_2^1$ or $t_1^1 \to t_{n-1}^0$. Both arcs result in a trail from w to Y without converging connections, and therefore lead to contradictions. Indeed, we find the trails

$$w \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to t_2^1 \rightleftharpoons \cdots \rightleftharpoons y,$$
$$w \rightleftharpoons \cdots \rightleftharpoons t_{n-1}^0 \leftarrow t_1^1 \rightleftharpoons \cdots \rightleftharpoons y,$$

which are better than (28).

Because the arc $c_1 \to y$ leads to a contradiction, we can assume that $c_1 \leftarrow y$. In this case the $\tilde{y}$ whose existence has been established from (v) cannot be equal to y since this would provide the cycle

$o \to y \to c_1 \to o$, and therefore a contradiction. Thus, G contains the subgraph below.



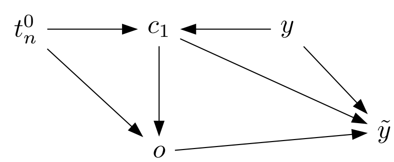
Here, we have $t_n^0 \in B(c_1, t_{n-1}^0)$ and $y \in B(c_1, \tilde{y})$. Similarly this means that $t_n^0 \to \tilde{y}$ or $y \to t_{n-1}^0$. Both arcs result in a trail from w to Y without converging connections, and therefore contradictions. Indeed, we find the trails

$$w \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to \tilde{y},$$
$$w \rightleftharpoons \cdots \rightleftharpoons t_{n-1}^0 \leftarrow y,$$

where the node $\tilde{y}$ is in Y by (v). This gives us the existence of the trail as claimed.

Case 3: Subgraph 12c.

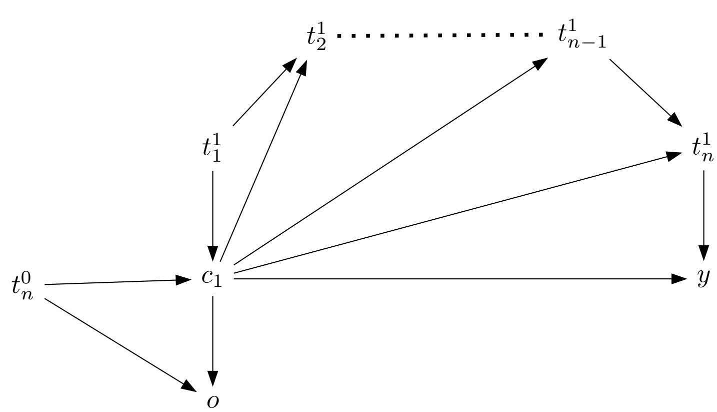
We must consider the two cases $c_1 \leftarrow y$ and $c_1 \to y$. First, let us assume that $c_1 \leftarrow y$, giving us the subgraph below. Again, $\tilde{y}$ cannot be equal to y, since this would create a cycle. Therefore, G contains the subgraph below.



Here, we have $t_n^0 \in B(c_1, o)$ and $y \in B(c_1, \tilde{y})$. Therefore, E must contain $t_n^0 \to \tilde{y}$ or $y \to o$. The former arc results in a trail

$$w \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to \tilde{y}$$

from w to Y without converging connections (and therefore a better trail than (28)) and the latter would contradict $y \notin pa(o)$ (which we know by (iv)). This means that $c_1 \to y$. Therefore, by combining subgraph 12c with Theorem A.4(iv) we obtain the subgraph below.



Here, we have $t_n^0 \in B(c_1, o)$ and $t_1^1 \in B(c_1, t_2^1)$. Therefore, we have $t_n^0 \to t_2^1$ or $t_1^1 \to o$. The arc $t_n^0 \to t_2^1$ provides the trail

$$w \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to t_2^1 \rightleftharpoons \cdots \rightleftharpoons y$$

between w and Y without converging connections, and thus a contradiction with the definition of (28). The arc $t_1^1 \to o$ provides us with the trail

$$w \to o \leftarrow t_1^1 \rightleftharpoons \cdots \rightleftharpoons t_n^1 \to y \tag{31}$$

which is better than the trail (28) according to $<_{TRAIL}$, unless $n_t(0) = 0$. In that case, (31) is also a minimal trail in $TRAILS\big(X, Y \,\big|\, Z\big)$ with one converging node which is equal to o. Therefore, we can apply the same argument as in Case 1 to the trail (31), which leads to a contradiction.

C.2 Possible candidates by outgoing arc

Assumption C.2. *Assumption A.1 holds. Furthermore, $w \in B(O_v^k) \setminus O_v^k$ and $o \in O_v^k$ are nodes such that (i) $o \to w$, (ii) $pa(w \downarrow o) \setminus O_v^k = \emptyset$, (iii) There is no arc from $B(O_v^k) \setminus O_v^k$ to O_v^k.*

Lemma C.3. *Assume that Assumption C.2 holds and that $w \notin PossCandOut(O_v^k)$. Then there exists a trail from w to a node in $O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\})$ which is activated by $pa(w \downarrow o) \sqcup \{o\}$ and contains no converging connections.*

Proof. By assumption we have $w \notin PossCandOut(O_v^k)$. Therefore, w is not a possible candidate by the outgoing arc $o \to w$. By the definition of a possible candidate by outgoing arc (see (10)), this means that one of the following conditions must be violated.

1. $pa(w \downarrow o) \subseteq O_v^k$.

2. $\mathrm{dsep}\big(w, O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\}) \,\big|\, pa(w \downarrow o) \sqcup \{o\}\big)$.

The first condition is satisfied, because $pa(w \downarrow o) \setminus O_v^k = \emptyset$ by Assumption C.2. Therefore, the second restriction must be violated, i.e. $\cancel{\mathrm{dsep}}\big(w, O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\}) \,\big|\, pa(w \downarrow o) \sqcup \{o\}\big)$. This means that there exists a trail from w to $O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\})$ activated by $pa(w \downarrow o) \sqcup \{o\}$.

Consequently, the set $TRAILS\big(X, Y \,\big|\, Z\big)$ with $X = \{w\}$, $Y = O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\})$ and $Z = pa(w \downarrow o) \sqcup \{o\}$ is not empty, and thus we can pick a minimal trail in this set:

$$w \rightleftharpoons \cdots \to c_1 \leftarrow \cdots\cdots\cdots \to c_C \leftarrow \cdots \rightleftharpoons y. \tag{32}$$

By Lemma C.4(v), such a trail has no converging connection, and this completes the proof of this lemma. $\square$

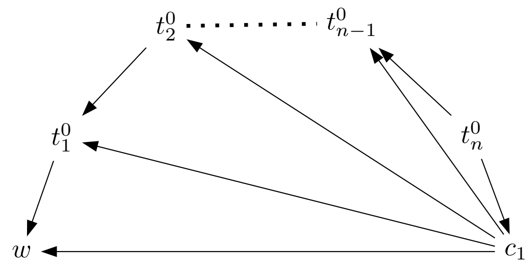
Lemma C.4. *Assume that Assumption C.2 holds and take a minimal trail*

$$w \rightleftharpoons \cdots \to c_1 \leftarrow \cdots\cdots\cdots \to c_C \leftarrow \cdots \rightleftharpoons y$$

in the set $TRAILS\big(X, Y \,\big|\, Z\big)$ with $X = \{w\}$, $Y = O_v^k \setminus (pa(w \downarrow o) \sqcup \{o\})$ and $Z = pa(w \downarrow o) \sqcup \{o\}$. Then, the following statements hold:

(i) $c_1 = Z(c_1)$*, where* $Z(c_i)$ *denotes the closest descendant of* c_i *in the sense of Definition A.5.*

(ii) The graph contains the subgraph below.

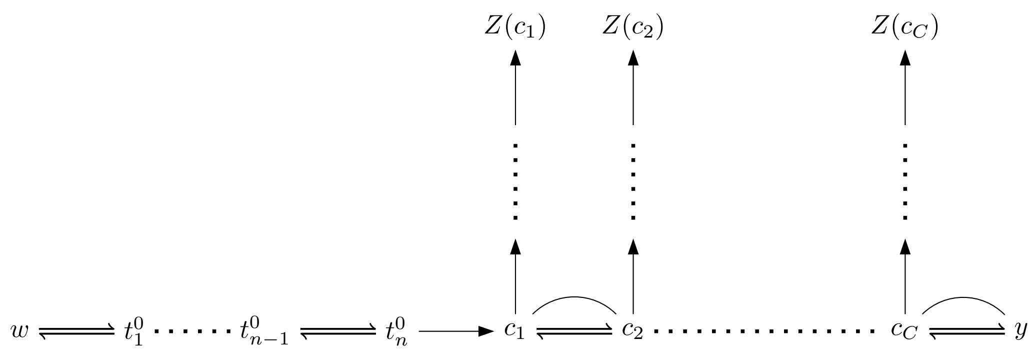


(iii) $c_1 \leftarrow c_2$.

(iv) For all $i = 1, \dots, C$*, we have that* $c_i = Z(c_i)$ *and* $c_i \leftarrow c_{i+1}$.

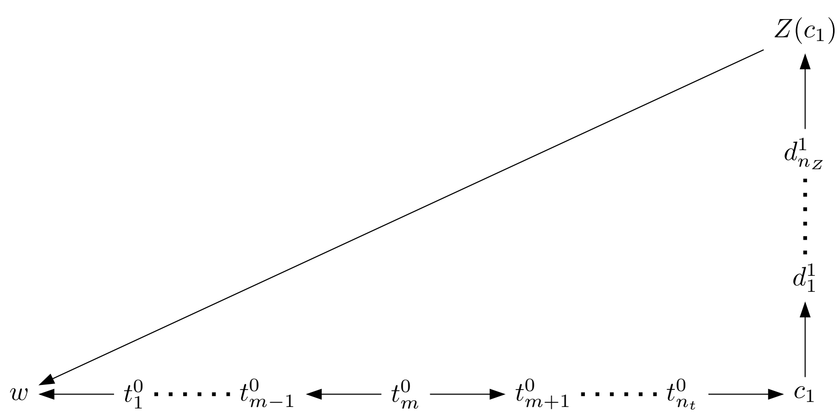
(v) The number of converging connections is equal to zero, i.e. $C = 0$.

Proof. First, note that $Y \sqcup Z = O_v^k$ have local relationships in the sense of Definition A.6, by Corollary B.2. Therefore, by Theorem A.4(iv), G contains the subgraph below with $y \in Y$.



C.2.1 Proof of Lemma C.4(i)

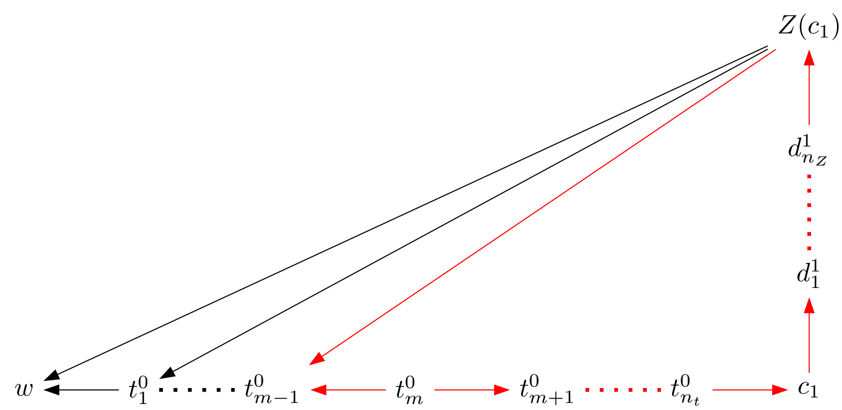
Let us assume that $c_1 \neq Z(c_1)$. Since $Z = pa(w \downarrow o) \sqcup \{o\}$, we have that $Z(c_i) \to w$ for all $i = 1, \dots, C$. In particular, we get that $Z(c_1) \to w$, and therefore G contains the subgraph below.



The undirected cycle above is an active cycle, unless the appropriate chords are present in G. Several chords can be excluded:

- The trails $c_1 \to d_1 \to \cdots \to d_{n_t} \to Z(c_1)$ and $w \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_{n_t}^0$ do not contain any chords by Theorem A.3(ii).
- $\forall l = 0, \ldots, n_t - 1$, $t_l^0 \to c_1$ results in a shorter trail (and $t_{n_t}^0 \to c_1$ is not a chord).
- $\forall l = 1, \ldots, n_t$, $t_l^0 \to Z(c_1)$ results in a trail with less converging connections not in Z.
- $\forall j = 1, \ldots, n_Z$, $\forall l = 1, \ldots, n_t$, $t_l^0 \to d_j^1$ results in a trail with shorter descendant paths.
- $\forall j = 0, \ldots, n_Z + 1$, $\forall l = m, \ldots, n_t$, $d_j^1 \to t_l^0$ results in a cycle.
- $\forall j = 0, \ldots, n_Z$, $\forall l = 0, \ldots, m - 1$, $d_j^1 \to t_l^0$ result in a trail with less converging connections.
- $\forall j = 1, \ldots, n_Z$, $w \to d_j^1$ results in a cycle.
- $\forall l = 0, \ldots, m - 1$, $c_1 \to t_l^0$ results in a trail with less converging connections.

Therefore, the only remaining chords are $Z(c_1) \to t_l^0$ with $l = 1, \ldots, m - 1$. It is evident that all such arcs must be present to prevent the appearance of an active cycle in G, giving us the subgraph below.



The subgraph above contains an undirected cycle with one converging connection (at t_{m-1}^0), coloured in red. Because there are no more chords which could be present, this undirected cycle must be of length smaller than 4, see Definition 3.6. The undirected cycle consists of the nodes c_1, $Z(c_1)$, $t_{m-1}^0, t_m^0, \ldots, t_{n_t}^0$ and $d_1^1, \ldots, d_{n_Z}^1$; therefore it is of length $2 + n_t - (m - 1) + 1 + n_Z = n_Z + n_t - m + 4$. This means that $n_Z + n_t - m + 4 \leq 3$, and therefore $n_Z + n_t - m \leq -1$. The equality can only hold if $n_Z = 0$ and $m = n_t + 1$. This is not possible because t_m is a diverging connection, while $t_{n_t+1}^0 = c_1$ is a converging connection.

So, if $c_1 \neq Z(c_1)$, we have shown that G contains an active cycle, and therefore we have proven that $c_1 = Z(c_1)$.

C.2.2 Proof of Lemma C.4(ii)

We know that G contains the trail

$$w \rightleftharpoons t_1^0 \rightleftharpoons \cdots \rightleftharpoons t_n^0 \to c_1.$$

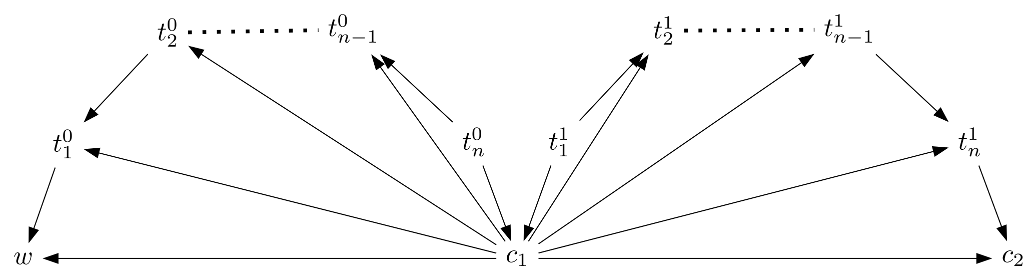
By (i), $c_1 \in Z = pa(w \downarrow o) \sqcup \{o\}$. Because this is a shortest trail activated by the empty set ending with a rightward arrow ($t_n^0 \to c_1$) consisting of nodes in $V \setminus Z$ and $c_1 = Z(c_1) \to w$ by definition of the set Z, we can apply Lemma A.1 and Theorem A.1 (with $v_1 = c_1$ and $v_2 = w$ in the notation of Theorem A.1) to find that G contains the subgraph as claimed.

Furthermore, the length n of this trail must be strictly larger than zero. If it were of length zero, then it would simply be the arc $w \to c_1$. However, this would result in a cycle, as we have shown that the arc $c_1 \to w$ must be present.

C.2.3 Proof of Lemma C.4(iii)

By Theorem A.4(iii), we know that c_1 and c_2 are adjacent. Therefore, it suffices to show that $c_1 \nrightarrow c_2$.

Suppose that $c_1 \to c_2$. Combining (ii) with Theorem A.4(iv) leads to the conclusion that G contains the subgraph below.



Here, we have that $t_n^0 \in B(c_i, t_{n-1}^0)$ and $t_1^1 \in B(c_1, t_2^1)$. Since G does not contain any interfering v-structures, we must have $t_1^1 \to t_{n-1}^0$ or $t_n^0 \to t_2^1$. However, both these arcs result in a better trail than (32). Indeed, the trails

$$w \leftarrow \cdots \leftarrow t_{n-1}^0 \leftarrow t_1^1 \rightleftharpoons \cdots\cdots\cdots y,$$
$$w \leftarrow \cdots \leftarrow t_n^0 \to t_2^1 \to \cdots\cdots\cdots \rightleftharpoons y,$$

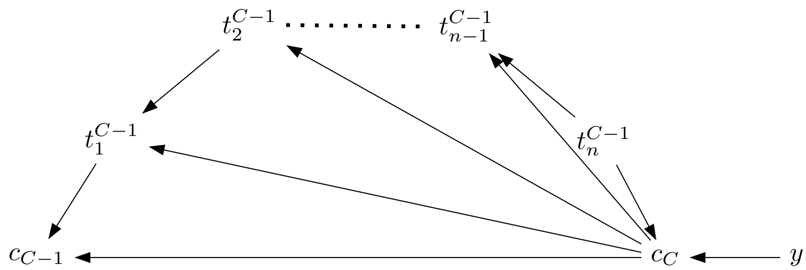
contain one fewer converging connection than (32), which gives a contradiction as claimed.

C.2.4 Proof of Lemma C.4(iv)

If $C = 1$, then the statement follows immediately by (i) and (iii). If $C > 1$, by (i) and (iii), we have that $c_1 = Z(c_1)$ and $c_1 \leftarrow c_2$. By Theorem A.4(iii), we know that for all $i = 1, \ldots, C$. c_i and c_{i+1} are adjacent By Lemma C.5, there cannot be any i such that $c_{i-1} \leftarrow c_i \to c_{i+1}$. Therefore, $i \in \{1, \ldots, C\}$, $c_i \leftarrow c_{i+1}$. Consequently, we can apply Corollary A.1(ii) to find that for all $i \in \{1, \ldots, C\}$, $c_i \in Z$, and therefore $c_i = Z(c_i)$.

C.2.5 Proof of Lemma C.4(v)

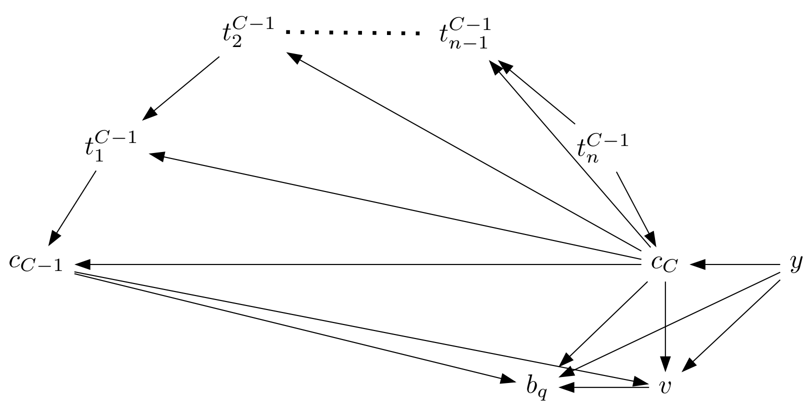
Assume that $C \geq 1$, then by (iv) we get $c_C \leftarrow y$. Combining this with Theorem A.4(iv), gives that G contains the subgraph below with the convention $c_0 := w$ in the case that $C = 1$.



We consider two cases; when $B(O_v^k) \neq pa(v)$ and when $B(O_v^k) = pa(v)$.

Case 1: Let us assume that $B(O_v^k) \neq pa(v)$ and let b_q be its corresponding node in the sense of Definition 4.4. Such a b_q always exists otherwise we would necessarily have $B(O_v^k) = pa(v)$. Remark that the nodes c_C and y are in $Y \sqcup Z = O_v^k$, and therefore they are in $B(O_v^k)$. Furthermore, if $C > 1$, then the node c_{C-1} is in $Z \subseteq O_v^k \subset B(O_v^k)$, and if $C = 1$, then $c_{C-1} = c_0 := w$ where w is in $B(O_v^k)$ by the assumptions of the lemma.

By Definition 4.4, any node in $B(O_v^k)$ has an arc pointing towards both v and b_q, giving us the subgraph below.



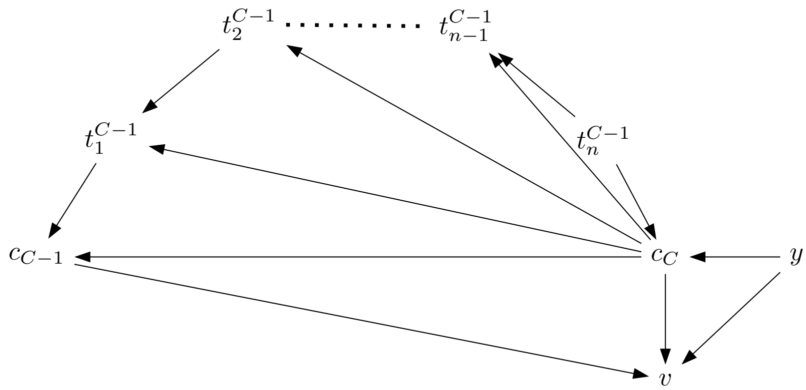
Here, we have that $t_n^{C-1} \in B(c_C, t_{n-1}^{C-1})$, $y \in B(c_C, v)$ and $y \in B(c_C, b_q)$. Since G does not contain any interfering v-structures, we must have $y \to t_{n-1}^{C-1}$, or both $t_n^{C-1} \to v$ and $t_n^{C-1} \to b_q$. The arc $y \to t_{n-1}^{C-1}$ results in a trail

$$w \rightleftharpoons \cdots \rightleftharpoons t_{n-1}^{C-1} \leftarrow y$$

with fewer converging connections than (32). By this contradiction, the arcs $t_n^{C-1} \to v$ and $t_n^{C-1} \to b_q$ must be present, and therefore $t_n^{C-1} \in pa(v) \cap pa(b_q) = B(v, b_q) = B_q$. Since $B_q = B(O_v^k)$, we obtain $t_n^{C-1} \in B(O_v^k)$. Moreover, by Theorem A.3(i) we know that $t_n^{C-1} \notin O_v^k$, and thus $t_n^{C-1} \in B(O_v^k) \setminus O_v^k$.

The arc $t_n^{C-1} \to c_C$ is now an arc from a node in $B(O_v^k) \setminus O_v^k$ to a node in O_v^k which is not possible by the assumptions of Lemma C.3. Therefore, this contradiction completes the proof of the case.

Case 2: If $B(O_v^k) = pa(v)$, then by a similar argument as in the first case we find that G must contain the subgraph



Remark that there are potential interfering v-structures at the nodes y and t_n^{C-1} to c_C. As in the previous case, we find that the arc $t_n^{C-1} \to v$ must be present. This means that $t_n^{C-1} \in pa(v) = B(O_v^k)$, and therefore by Theorem A.3(i) we have that $t_n^{C-1} \in B(O_v^k) \setminus O_v^k$. Hence, we again find the arc $t_n^{C-1} \to c_C$ from a node in $B(O_v^k) \setminus O_v^k$ to a node in O_v^k which is a contradiction.

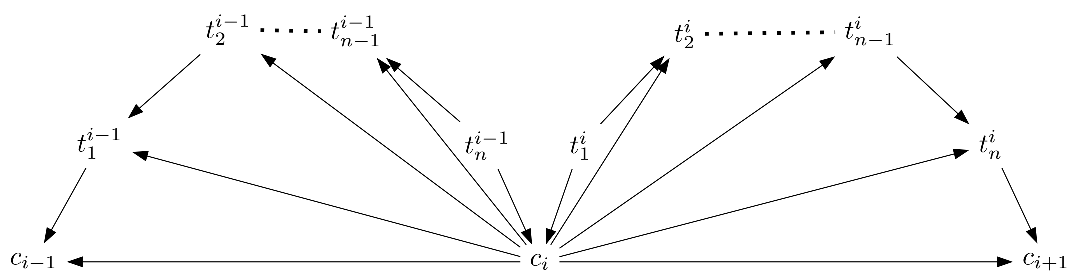
Thus, both cases are not possible when $C > 0$, which completes the proof of Lemma C.3. $\square$

C.3 Auxiliary lemmas for the proofs in Sections C.1 and C.2

Lemma C.5. *Assume that $G = (V, E)$ is a DAG with no interfering v-structures. Let $X, Y, Z \subseteq V$ be three disjoint subsets and $Y \sqcup Z$ has local relationships (Definition A.6). Assume that $TRAILS\big(X, Y \mid Z\big) \neq \emptyset$ and let T a trail of the form (22) be a minimal element of $TRAILS\big(X, Y \mid Z\big)$ with respect to the order $<_{TRAIL}$.*

Then for all $i = 2, \ldots, C$, the trail $c_{i-1} \leftarrow c_i \to c_{i+1}$ can not be present in G.

Proof. Suppose that there exists such a diverging connection. By Theorem A.4(iv), G contains the subgraph below.



Remark that

$$t_n^{i-1} \in B(c_i, t_{n-1}^{i-1}) = pa(c_i) \cap pa(t_{n-1}^{i-1}),$$
$$t_1^{i} \in B(c_i, t_2^{i}) = pa(c_i) \cap pa(t_2^{i}).$$

Since G does not contain any interfering v-structures, we must have $t_1^i \in B(c_i, t_{n-1}^{i-1})$ or $t_n^{i-1} \in B(c_i, t_2^i)$. This means that $t_1^i \to t_{n-1}^{i-1}$ or $t_n^{i-1} \to t_2^i$. However, both arcs result in the existence of trails between x and y

that have less converging connections than (22), and therefore are better trails than (22) whcih contradict assumptions. Hence, there cannot be diverging connection $c_{i-1} \leftarrow c_i \rightarrow c_{i+1}$, concluding the proof. $\square$

Lemma C.6. *Under Assumption A.1, let Y, Z be two subsets such that $Y \sqcup Z = O_v^k$, and let $y \in Y$. Consider the subgraph below where:*

- *The trail*
$$c_1 \leftarrow \cdots \rightarrow c_2 \leftarrow \cdots\cdots\cdots \rightarrow c_C \leftarrow \cdots \rightleftharpoons y$$
has C converging connections corresponding to the nodes $\{c_i\}_{i=1}^{C}$ with $C > 1$.
- *Each c_i is either contained in Z or it has a closest descendant $Z(c_i)$ in Z.*
- *All nodes on the trail and descendant paths not equal to y or $Z(c_i)$ with $i \in \{1, \ldots, C\}$ are in $V \setminus O_v^k$.*

Then, there exists a node $\tilde{o} \in O_v^k$ such that $y \rightarrow \tilde{o}$ and for all $i = 1, \ldots, C$, $Z(c_i) \rightarrow \tilde{o}$ whenever this does not result in the self-loop $\tilde{o} \rightarrow \tilde{o}$. Indeed, the node $\tilde{o}$ may be equal to any node in O_v^k including y and $Z(c_i)$ with $i \in \{1, \ldots, C\}$.

Proof. Remark that $\forall i = 1, \ldots, C$, $Z(c_i) \in Z \subseteq O_v^k$, and the node $y \in Y \subseteq O_v^k$. For convenience, we use the conventions $c_{C+1} := y$ and $Z(c_{C+1}) := y$. Therefore, the set $\{Z(c_i)\}_{i=1}^{C+1}$ must have a highest node according to the ordered set O_v^k. We denote such a node by o_{max} and pick $j \in \{1, \ldots, C+1\}$ such that $Z(c_j) = o_{max}$. This highest node o_{max} must have been a possible candidate to some partial order $\tilde{O} \subseteq O_v^k$ which contains all other nodes in the set, i.e. $\{Z(c_i) : i = 1, \ldots, C+1; i \neq j\}$.

Consequently, the node $Z(c_j)$ must be a possible candidate to a set $\tilde{O} \subseteq O_v^k$ which contains $\{Z(c_i) : i = 1, \ldots, C+1; i \neq j\}$. We now prove that it cannot be a candidate by independence. Observe that at least one of the following trails exist:

$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \rightarrow c_{j+1} \rightarrow \cdots \rightarrow Z(c_{j+1}),$$
$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \rightarrow c_{j+1} \rightarrow \cdots \rightarrow Z(c_{j-1}).$$

These are trails with no converging connections between $Z(c_j)$ and $\{Z(c_{j-1}), Z(c_{j+1})\} \subseteq \tilde{O}$ consisting of nodes in $V \setminus O_v^k \subseteq V \setminus \tilde{O}$. Therefore, we have that $\cancel{\text{dsep}}\big(Z(c_j), \tilde{O} \,\big|\, \emptyset\big)$, and therefore $Z(c_j) \notin PossCandInd(\tilde{O})$, see

Proposition 4.1. This means that $Z(c_j)$ must be in $PossCandIn(\tilde{O})$ or $PossCandOut(\tilde{O})$. We consider both cases.

Case 1: Suppose that $Z(c_j) \in PossCandIn(\tilde{O})$. Then, by Proposition 4.1, there exists an $\tilde{o} \in \tilde{O}$ such that $Z(c_j) \to \tilde{o}$ satisfying

1. $pa(\tilde{o} \downarrow Z(c_j)) \subseteq \tilde{O}$,
2. $\text{dsep}\big(Z(c_j), \tilde{O} \setminus (pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}) \,\big|\, pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}\big)$.

We will show that this implies that for all $i \neq j$, $Z(c_i) \in pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$. This means that each $Z(c_i)$ with $i \neq j$ points towards $\tilde{o}$ or is equal to $\tilde{o}$. Moreover, by the construction above we also know that $Z(c_j) \to \tilde{o}$. This finishes the proof of Lemma C.6.

Consider the nodes $Z(c_{j-1})$ and $Z(c_{j+1})$ (assuming that they exist). Suppose that the nodes $Z(c_{j-1})$ and $Z(c_{j+1})$ are not in $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$. They are connected to $Z(c_j)$ by the trails

$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j+1} \to \cdots \to Z(c_{j+1}),$$
$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j-1} \to \cdots \to Z(c_{j-1}),$$

which contain no converging connections nor nodes in $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\} \subseteq \tilde{O} \subseteq O_v^k$. Therefore, these trails are activated by $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$. Thus, they are trails from $Z(c_j)$ to $\tilde{O} \setminus (pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\})$ activated by $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$. This means that $\cancel{\text{dsep}}\big(Z(c_j), \tilde{O} \setminus (pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}) \,\big|\, pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}\big)$ which contradicts the assumption that the second restriction is satisfied. Therefore, we must have $Z(c_{j-1}), Z(c_{j+1}) \in pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$.

Now, the trails

$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j+1} \leftarrow \cdots \to c_{j+2} \to \cdots \to Z(c_{j+2}),$$
$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j-1} \leftarrow \cdots \to c_{j-2} \to \cdots \to Z(c_{j-2})$$

are activated by $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$ since the converging connections at c_{j-1} and c_{j+1} have a descendant ($Z(c_{j-1})$ and $Z(c_{j+1})$, respectively) in $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$. Thus, by the same argument $Z(c_{j-2})$ and $Z(c_{j+2})$ are in $pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$.

We conclude this proof by induction. Indeed, the same argument can be repeated to show that for any k, $Z(c_{j+k}) \in pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$ (resp. $Z(c_{j-k}) \in pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$) whenever $1 \leq j + k \leq C + 1$ (resp. $1 \leq j - k \leq C + 1$). Therefore, we have proved that for all $i \neq j$, $Z(c_i) \in pa(\tilde{o} \downarrow Z(c_j)) \sqcup \{\tilde{o}\}$, which completes the proof in this case.

Case 2: Suppose that $Z(c_j) \in PossCandOut(\tilde{O})$. Then, there exists an $\tilde{o} \in \tilde{O}$ with $\tilde{o} \to Z(c_j)$ satisfying

1. $pa(Z(c_j) \downarrow \tilde{o}) \subseteq \tilde{O}$,
2. $\text{dsep}\big(Z(c_j), \tilde{O} \setminus (pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}) \,\big|\, pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}\big)$.

We will show that for all $i \neq j$, $Z(c_i) \in pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$. Therefore, $i \neq j$, $Z(c_i) \to Z(c_j) \in O_v^k$, so $Z(c_j)$ is our desired $\tilde{o}$. This finishes the proof of C.6 in this case.

First, consider the nodes $Z(c_{j+1})$ and $Z(c_{j-1})$ (assuming that they exist). If they are both in $\tilde{O} \setminus (pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\})$, then

$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j+1} \leftarrow \cdots \to Z(c_{j+1}),$$
$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j-1} \leftarrow \cdots \to Z(c_{j-1}),$$

contain no converging connections nor nodes in $pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\} \subseteq O_v^k$. Therefore, they are activated by $pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$. This means that $\cancel{\mathrm{dsep}}\big(Z(c_j), \tilde{O} \setminus (pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}) \,\big|\, pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}\big)$ which contradicts the assumption that the second restriction is satisfied. Therefore, we must have $Z(c_{j-1}), Z(c_{j+1}) \in pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$.

Now, the trails

$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j+1} \leftarrow \cdots \to c_{j+2} \to \cdots \to Z(c_{j+2}),$$
$$Z(c_j) \leftarrow \cdots \leftarrow c_j \leftarrow \cdots \to c_{j-1} \leftarrow \cdots \to c_{j-2} \to \cdots \to Z(c_{j-2}),$$

are activated by $pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$. Thus, by the same argument $Z(c_{j+2}), Z(c_{j-2}) \in pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$.

Similarly as in the first case, the proof is finished by an induction argument, showing that for all $i \neq j$, $Z(c_i) \in pa(Z(c_j) \downarrow \tilde{o}) \sqcup \{\tilde{o}\}$, as claimed above. $\square$