ABSTRACT. We consider the problem of representing causal models that encode context-specific information for discrete data using a proper subclass of staged tree models which we call CStrees. We show that the context-specific information encoded by a CStree can be equivalently expressed via a collection of DAGs. As not all staged tree models admit this property, CStrees are a subclass that provides a transparent, intuitive and compact representation of context-specific causal information. We prove that CStrees admit a global Markov property which yields a graphical criterion for model equivalence generalizing that of Verma and Pearl for DAG models. These results extend to the general interventional model setting, making CStrees the first family of context-specific models admitting a characterization of interventional model equivalence.

We also provide a closed-form formula for the maximum likelihood estimator of a CStree and use it to show that the Bayesian information criterion is a locally consistent score function for this model class.

The performance of CStrees is analyzed on both simulated and real data, where we see that modeling with CStrees instead of general staged trees does not result in a significant loss of predictive accuracy, while affording DAG representations of context-specific causal information.

1. Introduction

In this paper we introduce CStrees and interventional CStrees to represent context-specific conditional independence relations among discrete random variables in directed acyclic graph (DAG) models and their interventional extensions.

1.1. CStrees. In a DAG model, the nodes of the graph correspond to random variables and the non-edges encode conditional independence (CI) relations that the data-generating distribution must entail. For instance, any joint distribution for the vector of random variables \((X_1, \ldots, X_p)\) in the model defined by a DAG \(G = ([p], E)\) satisfies the ordered Markov property which states that for some linear ordering \(\pi = \pi_1 \cdots \pi_p\) of the vertices \([p]\) that respects the directions of all arrows in \(G\), each node \(X_{\pi_i}\) is independent of all of preceding variables \(X_{\{\pi_1, \ldots, \pi_{i-1}\}\setminus \text{pa}_G(i)}\) given its direct causes \(\text{pa}_G(i)\); i.e., \(X_{\pi_i} \perp \perp X_{\{\pi_1, \ldots, \pi_{i-1}\}\setminus \text{pa}_G(i)} \mid X_{\text{pa}_G(i)}\). An intrinsic property of conditional independence statements of this form is that the independence of the two sets of variables \(X_{\pi_i}\) and \(X_{\{\pi_1, \ldots, \pi_{i-1}\}\setminus \text{pa}_G(i)}\) holds for any realization of the variables in the conditioning set \(X_{\text{pa}_G(i)}\). This feature of DAG models makes it infeasible to use them to encode distributions that satisfy context-specific conditional independence (CSI) relations; that is, conditional independence relations that hold only for a possibly strict subset of the realizations of the conditioning variables.

The problem of encoding additional CSI relations for distributions that belong to DAG models has been studied by several authors. These include the use of conditional probability tables (CPTs) with extra regularity structure [Boutilier et al., 1996], Bayesian multinets [Geiger and Heckerman, 1996], similarity networks [Heckerman, 1990] and chain event graphs (CEGs) [Smith and Anderson, 2008]. Each of these proposed models has their individual strengths: CEGs admit a high level
of expressiveness of context-specific information that comes at the cost of representations from which it is more difficult to extract causal information. On the other hand, Bayesian multinets are highly readable but limit their expressiveness to modeling how context-specific information impacts selected hypothesis variables. The contribution of CStrees is a family of causal models with the readability of a Bayesian multinet but with the ability to capture much more diverse CSI relations, similar to a CEG. CStrees are such that they admit a global Markov property and graphical characterization of model equivalence directly generalizing that of DAGs. These properties conveniently extend to interventional models, making CStrees the first family of context-specific causal models with a known graphical criterion for interventional model equivalence.

As an example, consider the staged tree in Figure 1a and its CEG representation in Figure 1b. From these two figures it might be difficult to read cause-effect relationships among variables as compared to a DAG, in which such information is encoded by a single directed edge between two nodes. Fortunately, the staged tree in Figure 1a is also a CStree. This means that the same model can be represented with the sequence of DAGs in Figure 1c, where the direct cause-effect relations in each context are evident. This representation is similar to a Bayesian multinet or similarity network but strictly more general in regards to the contexts in which causal information can be encoded. It is also worth noting that any distribution faithful to the tree in Figure 1a is only Markov to a complete DAG on four nodes. Hence, the alternative representation in Figure 1c is more expressive than a DAG while more transparent (i.e., easily read) than a staged tree or CEG.

Our point of departure is the theory of staged tree models, where CStrees admit a simple definition. After presenting the necessary background on staged trees, DAG models and CSI relations in Section 2, we devote Section 3 to establishing the main properties of CStrees. We show in Theorem 2.1 that a CStree model is equivalently represented by a collection of DAGs. Aside from this representation being more concise and readable, it can also be used to characterize CStrees that encode the same set of CSI relations (see Theorem 3.3). In particular, Corollary 3.6 characterizes equivalence of CStree models in terms of the skeletons and v-structures of their associated DAGs.

Figure 1. A CStree $\mathcal{T}$ on variables $X_1, X_2, X_3, X_4$, where $X_i$ has outcomes $\{x_i^1, x_i^2\}$. The same model can also be represented with the sequence of DAGs in (c).
We conclude Section 3 by deriving a closed-form formula for the maximum likelihood estimator of a CSTree (Proposition 3.7), and showing that the BIC is a locally consistent scoring criterion for CSTrees (Proposition 3.8). We also give a brief discussion on the enumerative properties of CSTrees.

1.2. Interventional CSTrees. It is well-known that two different DAGs can encode the same set of conditional independence statements implied by their, respective, ordered Markov properties. With CSTrees the same situation can occur, i.e., two different sets of context DAGs can encode the same CSI relations. Hence, the rudimentary causal information given via CI or CSI relations in the ordered Markov property is not enough to determine the directions of all direct cause-effect relations. A typical way of deciding on the direction of arrows that reverse between different, but equivalent, DAGs is to use additional data drawn from interventional distributions [Hauser and Bühlmann, 2012, Wang et al., 2017, Yang et al., 2018]. Using a combination of observational and interventional distributions it is possible to refine an equivalence class of DAGs into smaller, interventional equivalence classes [Yang et al., 2018]. Motivated by these developments, we introduce interventional CSTrees and show that they behave in a similar fashion to interventional DAG models. This helps to refine the equivalence classes of CSTrees by using interventions and it allows for the representation of more general context-specific interventional experiments.

Building on the characterization of equivalence for CSTrees provided in Section 3, we consider context-specific, general interventions in CSTrees in Section 4. We define interventional CSTrees models and formulate a context-specific I-Markov property (see Definition 4.2) that generalizes the I-Markov property for DAGs studied in [Yang et al., 2018]. Via this generalization, we extend the characterization of interventional equivalence of DAGs given in [Yang et al., 2018] to a characterization of equivalence of interventional CSTree models. To the best of the authors’ knowledge, this makes CSTrees the first family of context-specific interventional models admitting a characterization of interventional equivalence classes. These results are based on the theory of general interventions in staged tree models studied in [Duarte and Solus, 2020] as it applies to the class of CSTrees. In Sections 5 and 6 we apply these results to simulated and real data sets.

The proofs of all lemmas, theorems as well as supplementary material related to selected topics can be found in the appendices.

2. Preliminaries

In this section we introduce the necessary notation, definitions and theorems for staged tree models, discrete DAG models and context-specific conditional independence relations. For any positive integer \(m\), set \([m] := \{1, \ldots, m\}\). Let \(X_{[p]} \in R = (X_1, \ldots, X_p)\) be a vector of discrete random variables where \(X_i\) has state space \([d_i]\). The set \(\mathcal{R} := \prod_{i \in [p]}[d_i]\) is the state space of \(X_{[p]}\), and its elements are denoted by \(x_1 \cdots x_p, (x_1, \ldots, x_p)\) or simply by \(x\). When it is important to distinguish the variable and the outcome of the variable, we will represent element \(j \in [d_i]\) by \(x^j_i\). Given \(S \subseteq [p]\), we let \(X_S := (X_i : i \in S)\) be the subvector of \(X_{[p]}\) indexed by the elements in \(S\). The outcome space of \(X_S\) is \(\mathcal{R}_S := \prod_{i \in S}[d_i]\) and its elements are denoted by \(x_S\). Given two disjoint subsets \(S, T \subseteq [p]\), we let the concatenation of strings \(x_Sx_T\) or of vectors \((x_S, x_T)\) denote the corresponding realization of \(X_{S,T}\).

Let \(\mathbb{R}^{[\mathcal{R}]}\) denote the real Euclidean space with one standard basis vector \(e_x\) for each outcome \(x \in \mathcal{R}\). The probability density function \(f\) for a positive distribution over \(X_{[p]}\) can be realized as the point \((f(x) : x \in \mathcal{R})\) in the (open) probability simplex

\[
\Delta_{|\mathcal{R}|-1}^0 := \left\{ (f_x : x \in \mathcal{R}) \in \mathbb{R}^{[\mathcal{R}]} : f_x > 0 \text{ for all } x \in \mathcal{R} \text{ and } \sum_{x \in \mathcal{R}} f_x = 1 \right\}.
\]

A statistical model of positive distributions with realization space \(\mathcal{R}\) is a subset of \(\Delta_{|\mathcal{R}|-1}^0\).
2.1. Staged tree models. The class of staged tree models was first introduced in [Smith and Anderson, 2008], we refer the reader to [Collazo et al., 2018] for a detailed introduction to this model class.

A rooted tree $\mathcal{T} = (V, E)$ is a directed graph whose skeleton is a tree and for which there exists a unique node $r$, called the root of $\mathcal{T}$, whose set of parents \( \text{pa}_\mathcal{T}(r) := \{ k \in V : k \to r \in E \} \) is empty. For $v \in V$, let $\lambda(v)$ denote the unique directed path from $r$ to $v$ in $\mathcal{T}$. We also let $E(v) := \{ v \to w \in E : w \in ch_\mathcal{T}(v) \}$, where $\text{ch}_\mathcal{T}(v) := \{ k \in V : v \to k \in E \}$ is the set of children of $v$.

Definition 2.1. Let $\mathcal{T} = (V, E)$ be a rooted tree, $\mathcal{L}$ a finite set of labels, and $\theta: E \to \mathcal{L}$ a labeling of the edges $E$. The pair $(\mathcal{T}, \theta)$ is a staged tree if

1. $|\theta(E(v))| = |E(v)|$ for all $v \in V$, and
2. for any two $v, w \in V$, $\theta(E(v))$ and $\theta(E(w))$ are either equal or disjoint.

When the labeling $\theta$ is understood from context, we simply refer to $\mathcal{T}$ as a staged tree. The vertices of a staged tree are partitioned into disjoint sets called stages such that $v, w \in V$ are in the same stage if and only if $\theta(E(v)) = \theta(E(w))$. The partition of $V$ into its stages is called a staging of $\mathcal{T}$. When we depict a staged tree, such as in Figure 1a, we will color two nodes the same color to indicate that they are in the same stage, with the convention that any white nodes are in a stage with cardinality one.

The space of canonical parameters of a staged tree $\mathcal{T}$ is the set

$$\Theta_\mathcal{T} := \left\{ x \in \mathbb{R}^{|\mathcal{L}|} : \forall e \in E, x_{\theta(e)} \in (0, 1) \text{ and } \forall v \in V, \sum_{e \in E(v)} x_{\theta(e)} = 1 \right\}.$$

Definition 2.2. Let $\mathcal{T}$ be the collection of all leaves of $\mathcal{T}$. The staged tree model $\mathcal{M}_{(\mathcal{T}, \theta)}$ is the image of the map $\psi_\mathcal{T}: \Theta_\mathcal{T} \to \Delta_{|\mathcal{T}|-1}$ where

$$\psi_\mathcal{T}: x \mapsto f := \left( \prod_{e \in E(\lambda(v))} x_{\theta(e)} \right)_{v \in \mathcal{T}}.$$ 

We say that $f \in \Delta_{|\mathcal{T}|-1}$ factorizes according to $\mathcal{T}$ if $f \in \mathcal{M}_{(\mathcal{T}, \theta)}$. Two staged trees $\mathcal{T}$ and $\mathcal{T}'$ are called statistically equivalent if $\mathcal{M}_{(\mathcal{T}, \theta)} = \mathcal{M}_{(\mathcal{T}', \theta')}$. 

Throughout this paper we work with trees that represent the outcome space of the vector of discrete random variables $X_{[p]}$ as a sequence of events. Given an ordering $\pi_1 \cdots \pi_p$ of $[p]$, we construct a rooted tree $\mathcal{T} = (V, E)$ where $V := \{ r \} \cup \bigcup_{j \in [p]} \mathcal{R}_{\{\pi_1, \ldots, \pi_j\}}$, and

$$E := \{ r \to x_{\pi_1} : x_{\pi_1} \in [d_{\pi_1}] \} \cup \{ x_{\pi_1} \cdots x_{\pi_{k-1}} \to x_{\pi_1} \cdots x_{\pi_k} : x_{\pi_1} \cdots x_{\pi_k} \in \mathcal{R}_{\{\pi_1, \ldots, \pi_k\}}, k \in [p-1] \}.$$

The level of a node $v \in V$ is the number of edges in $\lambda(v)$, and for $k \in \{0, \ldots, p\}$, the $k^{th}$ level of $\mathcal{T}$, denoted $L_k$, is the collection of all nodes $v \in V$ with level $k$. Note that $L_0 = \{ r \}$, so we usually ignore this level when referring to the levels of $\mathcal{T}$. For the trees defined above and $k > 0$, the $k^{th}$ level of $\mathcal{T}$ is $L_k = \mathcal{R}_{\{\pi_1, \ldots, \pi_k\}}$. Hence, we associate the variable $X_{\pi_k}$ with level $L_k$ and denote this association by $(L_1, \ldots, L_p) \sim (X_{\pi_1}, \ldots, X_{\pi_p})$. Given such a tree $\mathcal{T}$ with levels $(L_1, \ldots, L_p) \sim (X_{\pi_1}, \ldots, X_{\pi_p})$, we call the ordering $\pi = \pi_1 \cdots \pi_p$ the causal ordering of $\mathcal{T}$. These trees are also uniform; meaning that $|E(v)| = |E(w)|$ for any $v, w \in L_k$, for all $k \geq 0$. A staged tree is stratified if all its leaves have the same level and if any two nodes in the same stage are also in the same level.

For a uniform and stratified staged tree model $\mathcal{M}_{(\mathcal{T}, \theta)}$, the parameter values on the edges $E(v)$ for $v \in \mathcal{R}$ abide by the chain rule (see [Duarte and Solus, 2020, Lemma 2.1]). Namely, if $f \in \mathcal{M}_{(\mathcal{T}, \theta)}$ then for any $v = x_{\pi_1} \cdots x_{\pi_p} \in \mathcal{R}$ and $e = x_{\pi_1} \cdots x_{\pi_{k-1}} \to x_{\pi_1} \cdots x_{\pi_k}$ in $\lambda(v)$, it holds that $x_{\theta(e)} =$
A uniform, stratified staged tree $\mathcal{T}$ with levels $(L_1, \ldots, L_p) \sim (X_{\pi_1}, \ldots, X_{\pi_p})$ is called compatibly labeled if
\[
\theta(x_{\pi_1} \cdots x_{\pi_k} \rightarrow x_{\pi_1} \cdots x_{\pi_k-1} x_{\pi_k}) = \theta(y_{\pi_1} \cdots y_{\pi_k-1} \rightarrow y_{\pi_1} \cdots y_{\pi_k-1} x_{\pi_k})
\]
for all $x_{\pi_k} \in [d_{\pi_k}]$ whenever $x_{\pi_1} \cdots x_{\pi_k-1}$ and $y_{\pi_1} \cdots y_{\pi_k-1}$ are in the same stage. This condition ensures that edges emanating from $x_{\pi_1} \cdots x_{\pi_k-1}$ and $y_{\pi_1} \cdots y_{\pi_k-1}$ with endpoints corresponding to the same outcome $x_{\pi_k}$ of the next variable encode the invariance of conditional probabilities:
\[
f(x_{\pi_k} | x_{\pi_1} \cdots x_{\pi_k-1}) = f(x_{\pi_k} | y_{\pi_1} \cdots y_{\pi_k-1}).
\]

2.2. DAG models. An important subclass of compatibly labeled staged trees are DAG models, which can alternatively be represented via a directed acyclic graph (DAG) [Smith and Anderson, 2008]. Let $G = ([p], E)$ be a DAG on node set $[p]$ with set of directed edges $E$.

A joint distribution, or its density function $f$, over the vector $X[p]$ is Markov to $G$ if $f$ satisfies the recursive factorization property:
\[
f(X_1, \ldots, X_p) = \prod_{i \in [p]} f(X_i | X_{pa_G(i)}).
\]

Let $M(G)$ denote the collection of all densities that are Markov to $G$. As noted earlier, we consider $M(G)$ as a subset of $\Delta_{[p]-1}^\circ$. A distribution (or $f$) satisfies the global Markov property with respect to $G$ if, given disjoint $A, B, C \subset [p]$ with $A, B \neq \emptyset$, it entails the CI relation $X_A \perp_{G} X_B | X_C$ whenever $A$ and $B$ are $d$-separated in $G$ given $C$. The notion of $d$-separation is defined in [Pearl, 2000]. The following theorem states the model $M(G)$ is in fact a conditional independence model determined by the CI relations encoded via $d$-separation in the DAG $G$.

**Theorem 2.1.** A distribution $f \in \Delta_{[p]-1}^\circ$ is Markov to a DAG $G$ if and only if it satisfies the global Markov property with respect to $G$.

It is possible that two different DAGs encode the exact same set of conditional independence relations via the global Markov properties, and hence represent the same model. In this case, the DAGs are Markov equivalent and we say that they belong to the same Markov equivalence class. There are multiple graphical characterizations of Markov equivalence [Andersson and Perlman, 1997, Chickering, 1995, Verma and Pearl, 1990a].

**Theorem 2.2** (Verma and Pearl [1990a]). Two DAGs $G$ and $H$ are Markov equivalent if and only if they have the same skeleton and $v$-structures.

The skeleton of a DAG refers to its underlying, undirected graph, and a $v$-structure refers to any pair of edges in the DAG $i \rightarrow k$ and $k \leftarrow j$ for which there is no edge between $i$ and $j$ in the DAG. All discrete DAG models are also staged tree models [Smith and Anderson, 2008]. A general construction for representing a DAG model with a staged tree is given in [Duarte and Solus, 2020, Example 3.2] as well as in [Smith and Anderson, 2008]. We let $T_G$ denote the collection of all staged trees representing a DAG model.

Let $A, B, C,$ and $S$ be disjoint subsets of $[p]$. We say that $A$ and $B$ are contextually independent given $S$ in the context $X_C = x_C$ if
\[
f(x_A | x_B, x_C, x_S) = f(x_A | x_C, x_S)
\]
holds for all $(x_A, x_B, x_S) \in \mathcal{R}_A \times \mathcal{R}_B \times \mathcal{R}_S$ whenever $f(x_A, x_B, x_S) > 0$. We call this a context-specific conditional independence relation, or CSI relation, denoted $X_A \perp_{S} X_B | X_S, X_C = x_C$.

A number of approaches have been proposed in an effort to account for CSI relations via DAG models. In addition to the Bayesian multinet, similarity networks and CEGs mentioned in the introduction, [Boutilier et al., 1996, Poole and Zhang, 2003] developed methods for representing the conditional probability tables (CPTs) for a DAG model by including CSI relations so as to speed-up computations for probabilistic inference. In [Chickering et al., 1997, Friedman and Goldszmidt,
the authors utilized these methods to develop algorithms for learning a DAG model that is more representative of the context-specific independencies in the data-generating distribution.

3. CStrees

Staged tree models can encode a wide range of CSI relations. However, since the staged tree model representation depends on an underlying event tree that encodes the outcomes in the model, the complexity of this representation can increase rapidly leading to an exponential blow-up of the number of parameters. In response to this, Smith and Anderson [2008] introduced the chain event graph (CEG) associated to a staged tree model.

The CEG is a useful graphical summary of the different possible sequences of events that happen in the staged tree. Although the CEG is a simpler representation, its complexity is also directly dependent on the total number of possible unfoldings of events that occur in the model. As illustrated in Figure 1b, it can still be challenging to read causal information directly from a CEG, even for very few variables.

In this section we define CStree models and show that they admit an alternative representation as a collection of context DAGs. The representation of a CStree via context DAGs has two advantages over its CEG representation: First, even if the number of outcomes in the model increases, the context DAG representation can still be very compact. Second, using context DAGs to encode the model assumptions allows the practitioner to transparently represent the direct causes of a variable in different contexts, whereas such information is typically harder to decipher from a CEG. Thus, restricting the class of staged tree models to CStrees increases the interpretability of the model without drastically increasing the complexity of the representation. Moreover, the class of CStrees admits a new criterion of model equivalence that does not hold for general staged trees and which is a direct generalization of Theorem 2.2.

3.1. Definition of CStrees. There are two defining conditions that distinguish CStrees from general staged trees: (1) the underlying event tree of a CStree always represents the outcome space of a vector of discrete random variables, and (2) the possible stagings of a CStree are more restrictive than in a general staged tree.

Definition 3.1. Let $\mathcal{T} = (V, E)$ be a compatibly labeled staged tree with levels $(L_1, \ldots, L_p) \sim (X_1, \ldots, X_p)$. We say that $\mathcal{T}$ is a CStree if for every stage $S_i \subset L_{k-1}$, there exists $C_i \subset [k-1]$ and $x_{C_i} \in \mathcal{R}_{C_i}$ such that

\[
S_i = \bigcup_{y[k-1]\setminus C_i \in \mathcal{R}_{[k-1]\setminus C_i}} \{x_{C_i}y[k-1]\setminus C_i\}. 
\]

We denote the collection of all CStrees by $\mathcal{T}_{CS}$.

In other words, a CStree is a compatibly labeled staged tree for which the vertices in a given stage are all outcomes that agree in a fixed set of indices. According to Definition 3.1, the family of CStrees is a generalization of DAG models defined via a context-specific relaxation of the ordered Markov property for DAGs.

A CStree for which for all $k \in [p-1]$,

\[
C_i = pa_G(k) \text{ for all } S_i \subset L_{k-1} \text{ and for every } x_{pa_G(k)} \in \mathcal{R}_{pa_G(k)}
\]

there is a stage $S_i \subset L_{k-1}$ satisfying

\[
S_i = \bigcup_{y[k-1]\setminus pa_G(k) \in \mathcal{R}_{[k-1]\setminus pa_G(k)}} \{x_{pa_G(k)}y[k-1]\setminus pa_G(k)\}
\]

represents a DAG model (see [Duarte and Solus, 2021a, Proposition 2.2]). That is, a DAG model corresponds to a CStree in which every level $L_{k-1}$ contains one stage for each possible outcome of a fixed subset of the preceding variables; namely, $X_{pa_G(k)}$, the direct causes of $X_k$. Such a staging is a direct encoding of the ordered Markov property for $G$ with respect to the causal ordering $\pi$ of the tree. CStrees generalize this to a context-specific ordered Markov property: Instead of requiring
that each level \( L_k \) contains a stage consisting of all vertices that agree with a fixed outcome of a fixed subset of preceding variables, \( X_{\Pi} \), for each possible outcome of \( X_{\Pi} \), we only require that each stage in \( L_k \) is the set of all vertices that agree with a fixed outcome of a subset of preceding variables. In particular, the set \( \Pi \) can vary between stages in the same level, and not all outcomes of any one \( X_{\Pi} \) need to correspond to stages.

**Example 3.1.** An example of a CStree on four binary variables with causal ordering 1234 is depicted in Figure 1a. In this example the outcome \( x_f \) corresponds to an upward arrow and \( x_i \) to a downward arrow, these represent the realizations of \( X_i \). The four non-singleton stages are determined by the contexts \( X_2 = x_1^1 \) in green, \( X_{\{1,3\}} = x_1^1x_3^1 \) in purple, \( X_{\{1,3\}} = x_1^1x_3^1 \) in red, and \( X_{\{1,3\}} = x_1^1x_3^2 \) in orange. Each of these contexts encodes a CSI relation. For instance, \( X_2 = x_3^1 \) encodes \( f(x_3^1 | x_1^1x_2^1) = f(x_3^1 | x_1^1x_2^2) = f(x_3^1 | x_2^1) \).

**3.2. Markov properties of CStrees.** In this section we characterize the set of all CSI relations that are entailed by a given CStree (Theorem 3.5). To this end, we define a CSI model over a collection of discrete random variables and establish a list of CSI axioms. We start by making explicit the way in which a CStree encodes CSI relations.

**Lemma 3.1.** Let \( T = (V, E) \) be a CStree with levels \( (X_1, \ldots, X_p) \sim (L_1, \ldots, L_p) \) and stages \( (S_1, \ldots, S_m) \). Then, for any \( f \in M(T, \theta) \) and \( S_i \subseteq L_{k-1} \), we have that \( f \) entails \( X_k = X_{[k-1]|C_i} | X_{C_i} = x_{C_i} \), where \( C_i = \{ \ell : \text{all elements in } S_i \text{ have the same outcome for } X_\ell \} \) hence \( x_{C_i} := y_{C_i} \) for any \( y \in S_i \).

We refer to the contexts \( X_{C_i} = x_{C_i} \) in Lemma 3.1 as *stage-defining contexts* for \( T \). It follows that CStrees encode CSI relations \( X_A \perp X_B | X_S, X_C = x_C \), and thereby represent a family of context-specific conditional independence models. A *conditional independence model* \( J \) over a set of variables \( V \) is a collection of triples \( \langle A, B | S \rangle \) where \( A, B, \) and \( S \) are disjoint subsets of \( V \), where \( S \) can be empty and \( \emptyset, B | S \) and \( A, \emptyset | S \) are always included in \( J \). A DAG \( G = ([p], E) \) encodes the conditional independence model

\[
J(G) = \{ \langle A, B | S \rangle : A, B \text{ d-separated in } G \text{ given } S \}.
\]

The model \( J(G) \) is called a *graphoid* since it is closed under the conditional independence axioms, including the intersection axiom [Sadeghi and Lauritzen, 2014].

In an analogous fashion, we define a *context-specific conditional independence model* \( J \) over a collection of discrete variables \( V \) to be a set of quadruples \( \langle A, B | S, X_C = x_C \rangle \) where \( A, B, C, \) and \( S \) are disjoint subsets of \( V \), \( X_C = x_C \) is a realization of the variables in \( C \), \( S \) and \( C \) can be empty, and \( \emptyset, B | S, X_C = x_C \) and \( A, \emptyset | S, X_C = x_C \) are always included in \( J \). We call a context-specific conditional independence model \( J \) a *context-specific graphoid* if it satisfies the context-specific conditional independence axioms (CSI axioms):

1. symmetry. If \( \langle A, B | S, X_C = x_C \rangle \in J \) then \( \langle B, A | S, X_C = x_C \rangle \in J \).
2. decomposition. If \( \langle A, B \cup D | S, X_C = x_C \rangle \in J \) then \( \langle A, B | S, X_C = x_C \rangle \in J \).
3. weak union. If \( \langle A, B \cup D | S, X_C = x_C \rangle \in J \) then \( \langle A, B | S \cup D, X_C = x_C \rangle \in J \).
4. contraction. If \( \langle A, B | S \cup D, X_C = x_C \rangle \in J \) then \( \langle A, D | S, X_C = x_C \rangle \in J \) then \( \langle A, B \cup D | S, X_C = x_C \rangle \in J \).
5. intersection. If \( \langle A, B | S \cup D, X_C = x_C \rangle \in J \) and \( \langle A, S | B \cup D, X_C = x_C \rangle \in J \) then \( \langle A, B \cup S | D, X_C = x_C \rangle \in J \).
6. specialization. If \( \langle A, B | S, X_C = x_C \rangle \in J \) and \( T \subseteq S \) and \( x_T \in R_T \), then \( \langle A, B | S \setminus T, X_{T \cup C} = x_{\bar{C}} \rangle \in J \).
7. absorption. If \( \langle A, B | S, X_C = x_C \rangle \in J \), \( T \subseteq C \) for which \( \langle A, B | S, X_{C\setminus T} = x_{C\setminus T}, X_T = x_T \rangle \in J \) for all \( x_T \in R_T \), then \( \langle A, B | S \cup T, X_{C \setminus T} = x_{C \setminus T} \rangle \in J \).

Given a set of CSI relations \( J \), we call the collection of relations \( \overline{J} \) produced by applying any of the operations (1) – (7) repeatedly to elements of \( J \) the *context-specific closure of \( J \).* Let \( \overline{J(T)} \)
denote the context-specific closure of the relations associated to $T$ according to Lemma 3.1. We say that a distribution $f \in \Delta^o_{|T| - 1}$ is Markov to $T$ whenever $f$ entails all CSI relations in $\mathcal{J}(T)$. By Duarte and Görgen [2020, Theorem 3] it then follows that $f$ is Markov to $T$ if and only if $f \in \mathcal{M}(T, \theta)$. Moreover, by the absorption axiom, there exists a collection of contexts $\mathcal{C}_T := \{X_C = x_C\}$ for $T$ such that for any

$$X_A \perp X_B \mid X_S, X_C = x_C \in \mathcal{J}(T),$$

with $X_C = x_C \in \mathcal{C}_T$, there is no subset $T \subseteq C$ for which

$$X_A \perp X_B \mid X_{S\cup T}, X_{C\setminus T} = x_{C\setminus T} \in \mathcal{J}(T).$$

We call each such $X_C = x_C$ a minimal context for $T$. In some cases, a repeated use of the absorption axiom can lead to a CI statement $X_A \perp X_B \mid X_S \in \mathcal{J}(T)$. In this case, $C = \emptyset$, the empty context (i.e., the context in which the value of no variable in the system is fixed). By definition, if the empty context appears in $\mathcal{J}(T)$, then it is a minimal context. When the only minimal context for $T$ is the empty context then $T \in \mathcal{T}_G$; that is, $T$ is the staged tree of a DAG model. This can be seen, for example, via [Duarte and Solus, 2021a, Proposition 2.2]. When $\mathcal{C}_T = R_{\{i\}}$, for some $i \in [p]$, then $T$ encodes a hypothesis-specific Bayesian multinet in which the variable $X_i$ is the hypothesis. As the next example illustrates, the collection of minimal contexts $\mathcal{C}_T$ can be more diverse form than these two options.

**Example 3.2.** The staged tree $T$ depicted in Figure 1a is a CStree. There are precisely four stages that contain more than one vertex, and each of these stages yields a context-specific independence statement as described in Lemma 3.1. These four statements are

$$X_4 \perp X_2 \mid (X_1, X_3) = x_1^1 x_3^1, \quad X_4 \perp X_2 \mid (X_1, X_3) = x_1^2 x_3^2,$$

$$X_4 \perp X_2 \mid (X_1, X_3) = x_2^1 x_3^1, \quad \text{and} \quad X_3 \perp X_1 \mid X_2 = x_2^1,$$

where $R_{\{1\}} = \{x_1^1, x_1^2\}$, $R_{\{2\}} = \{x_2^1, x_2^2\}$, and $R_{\{3\}} = \{x_3^1, x_3^2\}$. Equivalently, $T$ encodes the context-specific CI relations

$$X_4 \perp X_2 \mid X_1, X_3 = x_3^1, \quad X_4 \perp X_2 \mid X_3, X_1 = x_1^1, \quad \text{and} \quad X_3 \perp X_1 \mid X_2 = x_2^1,$$

and has the collection of minimal contexts $\mathcal{C}_T = \{X_1 = x_1^1, X_2 = x_2^1, X_3 = x_3^1\}$. Since $\mathcal{C}_T$ is not the outcome space of one single variable, but instead a collection of realizations of three distinct variables, it follows that the staged tree model $\mathcal{M}(T, \theta)$ of $T$ is not encoded by a Bayesian multinet.

While not all CStrees are Bayesian multinets, we can similarly associate a sequence of DAGs to a given CStree. This association will be the basis of the global Markov property of CStrees. To define this sequence of DAGs, we will use the following lemma.

**Lemma 3.2.** Suppose that $X_A \perp X_B \mid X_S, X_C = x_C \in \mathcal{J}(T)$. Then either

1. $X_C = x_C \in \mathcal{C}_T$, or
2. $X_A \perp X_B \mid X_S, X_C = x_C$ is implied by the specialization of some $X_A \perp X_B \mid X_{S'}, X_{C'} = x_{C'} \in \mathcal{J}(T)$, where $X_{C'} = x_{C'} \in \mathcal{C}_T$.

Moreover, every $X_C = x_C \in \mathcal{C}_T$ is a subcontext of (at least) one stage-defining context.

It follows from Lemma 3.2, and the fact that the CSI axioms (1) – (5) commute with absorption, that $\mathcal{J}(T)$ is equal to the closure under specialization of the union of context-specific graphiods

$$\bigcup_{X_C = x_C \in \mathcal{C}_T} \mathcal{J}_{X_C = x_C},$$

where $\mathcal{J}_{X_C = x_C}$ consists of all relations in $\mathcal{J}(T)$ with context $X_C = x_C$.

Given a conditional independence model $\mathcal{J}$ over variables $X_1, \ldots, X_p$ and an ordering $\pi = \pi_1 \cdots \pi_p$ of $[p]$, the minimal I-MAP of $\mathcal{J}$ with respect to $\pi$ is the DAG $\mathcal{G}_\pi = ([p], E_\pi)$ where
\( \pi_i \rightarrow \pi_j \in E \) if and only if \( i < j \) and \( X_{\pi_i} \perp X_{\pi_j} \mid X_{\{\pi_1, \ldots, \pi_{j-1}\} \setminus \{\pi_i\}} \in \mathcal{J} \). Let \( \mathcal{G}_{X_C=x_C} \) be a minimal I-MAP of the collection of CI relations

\[
\{ X_A \perp X_B \mid X_S : X_A \perp X_B \mid X_S, X_C = x_C \in \mathcal{J}_{X_C=x_C} \}
\]

with respect to the ordering on \([p]\) \setminus \mathcal{C} \) induced by the causal ordering of \( \mathcal{T} \). We call \( \mathcal{G}_{X_C=x_C} \) a context graph for \( \mathcal{T} \) and let \( \mathcal{G}_{\mathcal{T}} := \{ \mathcal{G}_{X_C=x_C} \mid x_C \in \mathcal{C} \} \) denote its complete collection of (minimal) context graphs. The collection of minimal context graphs from the staged tree \( \mathcal{T} \) from Example 3.2 is depicted in Figure 1c.

We say that \( f \in \Delta^0_{[R]} \) is Markov to \( \mathcal{G}_T \) if \( f \) entails all CSI relations encoded by \( \mathcal{G}_T \); that is, if, for all \( X_C = x_C \in \mathcal{C}_T \), \( f \) entails \( X_A \perp X_B \mid X_S, X_C = x_C \) whenever \( A \) and \( B \) are \( d \)-separated in \( \mathcal{G}_{X_C=x_C} \) given \( S \). Let \( \mathcal{M}(\mathcal{G}_T) \) denote the collection of all \( f \in \Delta^0_{[R]} \) that are Markov to \( \mathcal{G}_T \).

The following theorem generalizes Theorem 2.1 to CStrees.

**Theorem 3.3.** Let \( \mathcal{T} \) be a CStree with levels \( (L_1, \ldots, L_p) \sim (X_1, \ldots, X_p) \). For a distribution \( f \in \Delta^0_{[R]} \), the following are equivalent:

1. \( f \) factorizes according to \( \mathcal{T} \),
2. \( f \) is Markov to \( \mathcal{G}_T \), and
3. for all \( X_C = x_C \in \mathcal{C}_T \),

\[
f(X[p]\mid C \mid X_C = x_C) = \prod_{k \in [p]\setminus C} f(X_k \mid X_{pa_{X_C=x_C}}(k), X_C = x_C).
\]

In particular, \( \mathcal{M}(\mathcal{G}_T) = \mathcal{M}(\mathcal{G}(\mathcal{T}, \theta)) \).

Theorem 3.3 states that the complete set of CSI relations implied by the staging of a CStree \( \mathcal{T} \) is encoded by the \( d \)-separation statements of the context graphs in \( \mathcal{G}_T \). Hence, in analogy to the theory of DAG models, we can say that \( f \) satisfies the global Markov property with respect to \( \mathcal{T} \) if \( f \in \mathcal{M}(\mathcal{G}_T) \). The observation in Theorem 3.3 that \( \mathcal{M}(\mathcal{G}_T) = \mathcal{M}(\mathcal{G}(\mathcal{T}, \theta)) \) implies that all CStree models admit a representation as a sequence of DAGs, a much more intuitive representation for context-specific causal information that afforded by the staged tree or its CEG. In the next subsection, we will see a further advantage of CStrees over previously studied context-specific causal models; namely, that they admit a graphical and easily stated characterization of model equivalence.

**Remark 3.1.** Given a CStree \( \mathcal{T} \), for each \( X_C = x_C \in \mathcal{C}_T \), we can construct the context-specific subtree of \( \mathcal{T} \), denoted \( \mathcal{T}_{X_C=x_C} = (V_{X_C=x_C}, E_{X_C=x_C}) \), by deleting all subtrees \( \mathcal{T}_{x_1 \ldots x_k} \) with root node \( x_1 \ldots x_k \) and all edges \( x_1 \ldots x_{k-1} \rightarrow x_1 \ldots x_{k+1} \) for which \( x_k \notin X_C \setminus \{k\} \), and then contracting the edges \( x_1 \ldots x_{k-1} \rightarrow x_1 \ldots x_{k-1} \cap X_C \setminus \{k\} \) of \( \mathcal{T} \) for all \( x_1 \ldots x_{k-1} \in \mathcal{R}_{[k-1]} \), for all \( k \in \mathcal{C} \). The single node resulting from this edge contraction is labeled \( x_1 \ldots x_{k-1} \cap X_C \setminus \{k\} \) and is in the same stage as \( x_1 \ldots x_{k-1} \cap X_C \setminus \{k\} \) in \( \mathcal{T} \). Hence, \( \mathcal{T}_{X_C=x_C} \) is a staged tree, where all other nodes inherit their staging directly from \( \mathcal{T} \). One might suspect \( f(X[p]\mid C \mid X_C = x_C) \) to factorize according to the tree \( \mathcal{T}_{X_C=x_C} \), but this is not true in general. However, notice that if \( f \in \mathcal{M}(\mathcal{G}(\mathcal{T}, \theta)) \), the unnormalized distribution \( f(X[p]\mid C, X_C) \) factorizes according to \( \mathcal{T}_{X_C=x_C} \). Theorem 3.3 implies that the normalized distribution \( f(X[p]\mid C \mid X_C = x_C) \) factorizes according to a staged tree with the same levels as \( \mathcal{T}_{X_C=x_C} \) whose stages in each level are a refinement of those in \( \mathcal{T}_{X_C=x_C} \). The refinement of the stages is specified by the arrows in \( \mathcal{G}_{X_C=x_C} \).

### 3.3. Statistical equivalence of CStrees.

Statistical equivalence classes of staged tree models were characterized in [Görgen and Smith, 2018] by using an algebraic invariant associated to any staged tree called the interpolating polynomial. Using this tool, they showed that two staged trees \( \mathcal{T} \) and \( \mathcal{T}' \) are statistically equivalent if and only if \( \mathcal{T} \) can be transformed into \( \mathcal{T}' \) by applying a sequence of so-called swap and resize operators. Unfortunately, the vast generality of the family of all staged trees results in these transformations being too broad, making it difficult to verify statistical equivalence via this characterization. A more constructive approach was proposed in
Görgen et al. [2018] using polynomial equivalence and computer algebra to find all staged trees in the polynomial equivalence class of a given staged tree. The characterizations of equivalence just mentioned are inadequate for CStrees because the equivalent staged trees they produce are generally not CStrees. To overcome this limitation we show that CStrees admit another characterization of statistical equivalence which is a direct generalization of Theorem 2.2. To prove this, we start with the following lemma:

**Lemma 3.4.** If $\mathcal{T}$ and $\mathcal{T}'$ are two statistically equivalent CStrees then their sets of minimal contexts are equal; that is, $\mathcal{C}_\mathcal{T} = \mathcal{C}_{\mathcal{T}'}$.

We can then prove Theorem 3.5, yielding a check for statistical equivalence of a pair of CStrees, that depends on comparing the minimal contexts and context DAGs of the two to determine if they are equivalent. This alternative is simpler than the transformational characterization of Görgen and Smith [2018], with which, to verify non-equivalence of a pair of trees, one would have to compute the entire equivalence class of one staged tree and then check that it does not contain the second tree.

**Theorem 3.5.** Two CStrees, $\mathcal{T}$ and $\mathcal{T}'$, are statistically equivalent if and only if they have the same set of minimal contexts and their minimal contexts graphs are pairwise Markov equivalent; that is, $\mathcal{C}_\mathcal{T} = \mathcal{C}_{\mathcal{T}'}$ and $\mathcal{G}_{X_C=x_C} \in \mathcal{G}_\mathcal{T}$ and $\mathcal{G}'_{X_C=x_C} \in \mathcal{G}_{\mathcal{T}'}$ are Markov equivalent for all $X_C = x_C \in \mathcal{C}_\mathcal{T}$.

Following the terminology for DAGs, we then say that two CStrees, $\mathcal{T}$ and $\mathcal{T}'$, are Markov equivalent if and only if they have the same set of minimal contexts and their contexts graphs are pairwise Markov equivalent. Theorem 3.5 allows us to extend the classic Markov equivalence result of Verma and Pearl [1990] to the family of CStrees.

**Corollary 3.6.** Two CStrees $\mathcal{T}$ and $\mathcal{T}'$ are statistically equivalent if and only if $\mathcal{C}_\mathcal{T} = \mathcal{C}_{\mathcal{T}'}$ and for all $X_C = x_C \in \mathcal{C}_\mathcal{T}$, the graphs $\mathcal{G}_{X_C=x_C} \in \mathcal{G}_\mathcal{T}$ and $\mathcal{G}'_{X_C=x_C} \in \mathcal{G}_{\mathcal{T}'}$ have the same skeleton and $v$-structures.

### 3.4. Maximum likelihood estimation for CStrees.

We now present a closed-form formula for the MLE of a CStree. We derive the formula from results about the MLE of a staged tree model in [Duarte et al., 2021]. In particular, the MLE of a staged tree model is an invariant of its statistical equivalence class.

We consider data $\mathbb{D}$ summarized as a $d_1 \times d_2 \times \cdots \times d_p$ contingency table $u$. The entry $u_{x}$ of $u$ is the number of occurrences of the outcome $x = x_1 x_2 \ldots x_p \in \mathcal{R}$ in $\mathbb{D}$. Given $C \subset [p]$ we consider the marginal table $u_C$. The entry $u_{x,C}$ in the table $u_C$ is obtained by fixing the indices of the states in $C$ and summing over all other indices not in $C$. That is,

$$u_{x,C} = \sum_{y \in \mathcal{R}_{[p] \setminus C}} u_{x,C,y}.$$

**Proposition 3.7.** Let $\mathcal{T}$ be a CStree with levels $(L_1, \ldots, L_p) \sim (X_1, \ldots, X_p)$. The MLE in $\mathcal{M}_{(\mathcal{T},\theta)}$ for the table $u$ is

$$\hat{p}_x = \prod_{k=1}^{p} \frac{u_{x,C_j \cup \{k\}}}{u_{x,C_j}},$$

where $x \in \mathcal{R}$ and $C_j$ is the stage defining context of $S_j$ that contains the node $x_{[k-1]}$. If $\mathcal{T}$ represents a DAG model, then $C_j = \text{pa}_{\mathcal{G}}(k)$.

Given data $\mathbb{D}$ drawn from a joint distribution over variables $(X_1, \ldots, X_p)$ and a DAG $\mathcal{G} = ([p], E)$, the *Bayesian Information Criterion* (BIC) is defined as

$$S(\mathcal{G}, \mathbb{D}) = \log p(\mathbb{D} \mid \hat{\theta}, \mathcal{G}) - \frac{d}{2} \log(n),$$
where \( \hat{\theta} \) denotes the maximum likelihood values for the DAG model parameters, \( d \) denotes the number of free parameters in the model, and \( n \) denotes the sample size. In a similar fashion, the \( \text{BIC} \) of a CStree \( T \) is defined as

\[
\mathcal{S}(T, \mathbb{D}) = \log p(\mathbb{D} \mid \hat{\theta}, T) - \frac{d}{2} \log(n),
\]

where the number of free parameters is the sum over \( k \in [p-1] \) of the product of \( |\mathcal{R}_{\{k\}}| - 1 \) and the number of distinct stages in level \( k \) of \( T \). For example, when all variables are binary, \( d \) is the number of stages in \( T \). By Corollary 3.6, the number of free parameters \( d \) is the same for any two statistically equivalent staged trees, as the stages in each tree are determined by the edges in their associated context graphs. Hence, the \( \text{BIC} \) is score equivalent for CStrees, meaning that \( \mathcal{S}(T, \mathbb{D}) = \mathcal{S}(T', \mathbb{D}) \) whenever \( T \) and \( T' \) are statistically equivalent. By Theorem 3.3, CStrees are examples of discrete DAG models with explicit local constraints, and hence are curved exponential models \citep{Geiger2001}. This observation also follows from a recent result of \cite{Gorgen2020} who showed that all staged tree models are curved exponential models. Thus, it follows from a result in \cite{Haughton1988}, that the \( \text{BIC} \) is consistent which means it satisfies the conditions in the next definition:

**Definition 3.2.** Let \( \mathbb{D} \) be a set of \( n \) independent and identically distributed samples drawn from some distribution \( f \in \Delta_{|\mathcal{X}| - 1}^p \). A scoring criterion \( \mathcal{S} \) is consistent if, as \( n \to \infty \), the following holds for any two models \( M, M' \subset \Delta_{|\mathcal{X}| - 1}^p \):

1. \( \mathcal{S}(M, \mathbb{D}) > \mathcal{S}(M', \mathbb{D}) \) whenever \( f \in M \) but \( f \notin M' \), and
2. \( \mathcal{S}(M, \mathbb{D}) > \mathcal{S}(M', \mathbb{D}) \) whenever \( f \in M \cap M' \) but \( M \) has fewer parameters than \( M' \).

A key feature of DAG models is that the \( \text{BIC} \) is not only consistent but locally consistent \citep{Chickering2002}. The definition of local consistency for DAG models can be naturally extended to CStrees.

**Definition 3.3.** Let \( \mathbb{D} \) be a set of \( n \) independent and identically distributed samples drawn from some distribution \( f \in \Delta_{|\mathcal{X}| - 1}^p \). Let \( T \) be a CStree with levels \( (L_1, \ldots, L_p) \) and suppose that \( T' \) is a CStree resulting from partitioning the stage \( S_i \subset L_{k-1} \) with associated context \( X_{C_i} = x_{C_i} \) according to the outcomes of \( X_j \) for some \( j < k \) with \( j \notin C_i \). A scoring criterion \( \mathcal{S} \) is locally consistent if the following holds:

1. If \( X_k \not\perp X_j \mid X_{\text{pa}_X_{C_i}}(k) \cdot X_C = x_C \) for every subcontext \( X_C = x_C \in \mathcal{C}_T \) of \( X_{C_i} = x_{C_i} \) then \( \mathcal{S}(T', \mathbb{D}) > \mathcal{S}(T, \mathbb{D}) \).
2. If \( X_k \perp X_j \mid X_{\text{pa}_X_{C_i}}(k) \cdot X_C = x_C \) for every subcontext \( X_C = x_C \in \mathcal{C}_T \) of \( X_{C_i} = x_{C_i} \) then \( \mathcal{S}(T', \mathbb{D}) < \mathcal{S}(T, \mathbb{D}) \).

Note by Lemmas 3.1 and 3.2 that Definition 3.3 (1) is equivalent to the condition that \( \mathcal{S}(T', \mathbb{D}) > \mathcal{S}(T, \mathbb{D}) \) whenever \( X_k \not\perp X_{[k-1]\setminus\{j\}} \mid X_{C_{\cup\{j\}}} = x_C x_j \) for some \( x_j \in [d_j] \). Similarly, Definition 3.3 (2) is equivalent to the condition that \( \mathcal{S}(T', \mathbb{D}) < \mathcal{S}(T, \mathbb{D}) \) whenever \( X_k \perp X_{[k-1]\setminus\{j\}} \mid X_{C_{\cup\{j\}}} = x_C x_j \) for all \( x_j \in [d_j] \). One consequence of the closed-form formula for the maximum likelihood estimate given in Proposition 3.7 is that BIC is also a locally consistent scoring criterion for CStrees.

**Proposition 3.8.** The BIC is a locally consistent scoring criterion for CStrees.

3.5. **On the enumeration of CStrees.** Similar to DAGs, the number of CStrees grows super-exponentially in the number of variables \( p \). The number of CStrees for 1, 2, 3, 4 and 5 binary variables is depicted side-by-side with the corresponding number of DAGs and compatibly labeled staged trees in Figure 2. We see that the number of CStrees for representing \( p \) binary variables reaches this order of magnitude around \( p = 7 \).
Figure 2. Number of DAGs, CStrees and compatibly labeled staged trees on $p$ binary variables.

general compatibly labeled staged trees on $p = 5$ binary variables is already on the order of 100’s of millions of billions. It is well-known [Cowell and Smith, 2014] and easy to verify that the number of compatibly labeled staged trees on $p$ binary variables is

$$p! \prod_{k=1}^{p-1} B_{2k}$$

where $B_p$ is the $p^{th}$ Bell number [OEIS, 2010, A000110]. A similar formula holds for the number of CStrees:

Let $[0,1]^p$ denote the $p$-dimensional cube that is given by the convex hull of all $(0,1)$-vectors in $\mathbb{R}^p$. We define the $(p+1)^{st}$ cubical Bell number to be the number of ways to partition the vertices of $[0,1]^p$ into non-overlapping faces of the cube. In Appendix A.8, we prove the following proposition and relate the numbers $B_p^{(c)}$ to the classical Bell numbers $B_p$.

**Proposition 3.9.** The number of CStrees on $p$ binary variables is $p! \prod_{k=1}^{p} B_{k}^{(c)}$.

The cubical Bell numbers are known only for small values [OEIS, 2010, A018926]. For $p = 1,\ldots,6$ they are $1,2,8,154,89512,71319425714$. The number of CStrees in Figure 2 are computed via these values and Proposition 3.9. Given Proposition 3.9, it would be of interest to have a closed-form formula for $B_p^{(c)}$.

4. Interventional CStrees

Assuming no additional constraints on the data-generating distribution, a model representative, such as a DAG or CStree, can only be distinguished up to statistical equivalence; that is, we can only determine its equivalence class of DAGs or CStrees. As we see from Theorem 2.2 and Corollary 3.6, the different members of a given statistical equivalence class can encode very different causal relationships amongst the variables. By incorporating additional data sampled from interventional distributions, one can further distinguish a causal structure from the other elements of its statistical equivalence class. This idea has become increasingly popular in relation to DAG models in the last decade [Hauser and Bühlmann, 2012, Kocaoglu et al., 2019, Wang et al., 2017, Yang et al., 2018].

Similarly, a theory for general interventions in staged trees was recently proposed in [Duarte and Solus, 2020], which generalizes the previous work of [Riccomagno and Smith, 2007, Thwaites, 2008, Thwaites et al., 2010]. In this section, we specialize this theory to the family of CStrees, where we can recover an interventional global Markov property and a characterization of interventional model equivalence that extend Theorem 3.3 and Corollary 3.6, respectively, to interventional CStrees. These results generalize the theory for interventional DAG models developed in [Yang et al., 2018] and give a framework to model interventions in context-specific DAGs.

| $p$ | DAGs | CStrees | Compatibly Labeled Staged Trees |
|-----|------|---------|---------------------------------|
| 1   | 1    | 1       | 1                               |
| 2   | 3    | 4       | 4                               |
| 3   | 25   | 96      | 180                             |
| 4   | 543  | 59136   | 2980800                         |
| 5   | 29281 | 26466908160 | 156196038558888000          |
| 6   | 3781503 | 1.1326 × 10^{22} | 1.20019 × 10^{44} |
| 7   | 1138779265 | ? | 1.44616 × 10^{110} |
| 8   | 783702329343 | ? | 1.29814 × 10^{269} |
4.1. Interventional DAG models. Let \( f \) be the density function of a distribution over random variables \( X_p \). Given a subset \( I \subset [p] \), called an intervention target, we say that a density \( f^{(I)} \) is an interventional density with respect to \( f, \mathcal{G} \) and \( I \) if it factors as

\[
f^{(I)}(X_1, \ldots, X_p) = \prod_{j \in I} f^{(I)}(X_j \mid X_{pa_G(j)}) \prod_{j \notin I} f(X_j \mid X_{pa_G(j)}).
\]

The density \( f \) is called the observational density. The key feature of an interventional density is the invariance of the conditional factors associated to the variables not targeted by \( I \). These invariance properties allow us to distinguish between elements of a Markov equivalence class by comparing data from the observational and interventional distributions [Pearl, 2000]. If the intervention eliminates the dependencies between a node and its parents in \( \mathcal{G} \), it is called a perfect or hard intervention [Eberhardt et al., 2005]. Otherwise, it is called a soft intervention. The term general intervention refers to an intervention that is either hard or soft. Since the observational density can be thought of as an interventional density for which the target set \( I \) is empty, we typically denote it by \( f^{(\emptyset)} \).

In practice, we may have more than one interventional target yielding an interventional distribution. Given a collection of interventional targets \( \mathcal{I} \) with \( \emptyset \in \mathcal{I} \), an interventional setting for \( \mathcal{I} \) and \( \mathcal{G} \) is a sequence of interventional densities \( (f^{(I)})_{I \in \mathcal{I}} \), where for each \( I \in \mathcal{I} \), \( f^{(I)} \) is an interventional density with respect to \( I, \mathcal{G} \) and \( f^{(\emptyset)} \). The collection of all such interventional settings is called the interventional DAG model for \( \mathcal{G} \) and \( \mathcal{I} \), and it is denoted \( \mathcal{M}_\mathcal{I}(\mathcal{G}) \). Note that

\[
\mathcal{M}_\mathcal{I}(\mathcal{G}) = \{(f^{(I)})_{I \in \mathcal{I}} \mid \forall I, J \in \mathcal{I} : f^{(I)} \in \mathcal{M}(\mathcal{G}) \text{ and } f^{(I)}(X_j \mid X_{pa_G(j)}) = f^{(J)}(X_j \mid X_{pa_G(j)}) \forall j \notin I \cup J \}.
\]

Yang et al. [2018] prove that the elements in \( \mathcal{M}_\mathcal{I}(\mathcal{G}) \) can be characterized via an interventional global Markov property associated to a DAG; namely, an \( \mathcal{I} \)-DAG associated to \( \mathcal{G} \) and \( \mathcal{I} \), where \( \mathcal{G}^\mathcal{I} := ([p] \cup W_\mathcal{I}, E \cup E_\mathcal{I}) \),

\[
W_\mathcal{I} := \{w_I \mid I \in \mathcal{I} \setminus \{\emptyset\}\} \quad \text{and} \quad E_\mathcal{I} := \{w_I \rightarrow j \mid j \in I, \forall I \in \mathcal{I} \setminus \{\emptyset\}\}.
\]

**Definition 4.1.** Let \( \mathcal{I} \) be a collection of intervention targets with \( \emptyset \in \mathcal{I} \). Let \( (f^{(I)})_{I \in \mathcal{I}} \) be a set of (strictly positive) probability densities over \( (X_1, \ldots, X_p) \). Then \( (f^{(I)})_{I \in \mathcal{I}} \) satisfies the \( \mathcal{I} \)-Markov property with respect to \( \mathcal{G} \) and \( \mathcal{I} \) if

1. \( X_A \perp X_B \mid X_C \) for any \( I \in \mathcal{I} \) and any disjoint \( A, B, C \subset [p] \) for which \( C \) d-separates \( A \) and \( B \) in \( \mathcal{G} \).
2. \( f^{(I)}(X_A \mid X_C) = f^{(\emptyset)}(X_A \mid X_C) \) for any \( I \in \mathcal{I} \) and any disjoint \( A, C \subset [p] \) for which \( C \cup W_\mathcal{I} \setminus \{I\} \) d-separates \( A \) and \( w_I \) in \( \mathcal{G}^\mathcal{I} \).

The next theorem uses the \( \mathcal{I} \)-Markov property to generalize Theorem 2.1 to interventional DAG models.

**Theorem 4.1** (Proposition 3.8, Yang et al. [2018]). Suppose that \( \emptyset \in \mathcal{I} \). Then \( (f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_\mathcal{I}(\mathcal{G}) \) if and only if \( (f^{(I)})_{I \in \mathcal{I}} \) satisfies the \( \mathcal{I} \)-Markov property with respect to \( \mathcal{G} \) and \( \mathcal{I} \).

Two DAGs \( \mathcal{G} \) and \( \mathcal{H} \) are called \( \mathcal{I} \)-Markov equivalent if \( \mathcal{M}_\mathcal{I}(\mathcal{G}) = \mathcal{M}_\mathcal{I}(\mathcal{H}) \). The following result generalizes the characterization of Verma and Pearl (Theorem 2.2) for DAGs.

**Theorem 4.2** (Theorem 3.9, Yang et al. [2018]). Two DAGs \( \mathcal{G} \) and \( \mathcal{H} \) are \( \mathcal{I} \)-Markov equivalent for \( \mathcal{I} \) containing \( \emptyset \) if and only if \( \mathcal{G}^\mathcal{I} \) and \( \mathcal{H}^\mathcal{I} \) have the same skeleton and v-structures.

It follows from Theorem 4.2 that \( \mathcal{I} \)-Markov equivalence classes refine Markov equivalence classes of DAGs, allowing for the representation of more refined causal relations. In subsection 4.4, we generalize Theorems 4.1 and 4.2 to analogous statements for interventional CStrees.
4.2. Interventional CStrees. Interventional staged trees were studied in [Duarte and Solus, 2020] for modeling general interventions in context-specific settings and to supply the machinery for generalizing an algebro-geometric characterization of decomposable models to the interventional setting [Duarte and Solus, 2021a, Geiger et al., 2006]. Consequently, the definition given in [Duarte and Solus, 2020] is extremely general, allowing for quite diverse forms of intervention in discrete systems. However, similar to staged trees, such models with high levels of expressiveness suffer from exponential blow-up in the number of parameters. In the following, our goal is to give a subfamily of these very general interventional staged trees that allow for more control over the number of parameters, similar to DAG models.

Let $T$ be a CStree with levels $(L_1, \ldots, L_p) \sim (X_1, \ldots, X_p)$ and let $S$ be a collection of sets of stages of $T$. Each set $S \in S$ can be thought of as an intervention target, but for the purposes of aligning our definition with the more general theory of interventional staged trees in [Duarte and Solus, 2020] we let

$$I = \{ \{ x_1 \cdots x_{k-1} x_k \in R_{[k]} : x_1 \cdots x_{k-1} \in T, T \in S, x_k \in R_{[k]} \} : S \in S \}.$$

be the set of intervention targets for our model. Note that $I$ contains a set of nodes that are precisely the children of all nodes in the stages in $S$ for each $S \in S$. In particular, equation (2) defines a bijection of sets $\psi : S \rightarrow I$ where

$$\psi : S \mapsto \{ x_1 \cdots x_{k-1} x_k \in R_{[k]} : x_1 \cdots x_{k-1} \in T, T \in S, x_k \in R_{[k]} \}$$

In the following, if $I = \{ I_0, \ldots, I_K \}$, we let $S_k \in S = \psi^{-1}(I)$ denote the set of stages $\psi^{-1}(I_k)$ for all $k = 0, \ldots, K$. Given $S \in S$ we also let $s := \cup_{T \in S} T$ denote the union of the stages in $S$. Similar, to DAGs, we define the interventional CStree model for $T$ and $I$ to be

$$M_I(T) := \{(f^{(I)})_{I \in I} : \forall I, J \in I, f^{(I)} \in M_{(T, \theta)} \text{ and } f^{(I)}(x_k \mid x_1 \cdots x_{k-1}) = f^{(J)}(x_k \mid x_1 \cdots x_{k-1}) \forall x_1 \cdots x_k \notin I \cup J \}.$$

We represent the model $M_I(T)$ using a rooted tree that is constructed as follows: Label $I = \{ I_0, \ldots, I_K \}$, and for each $k = 0, \ldots, K$ let $T^{(k)} = (V^{(k)}, E^{(k)})$ denote a copy of the staged tree $T = (V, E)$ where each $x_1 \cdots x_j \in V$ corresponds to a vertex $x^{(k)}_{1} \cdots x^{(k)}_{j} \in V^{(k)}$ and similarly for the edges $E^{(k)}$. Let $r^{(k)}$ denote the root of $T^{(k)}$ and define a new root node $r$. Connect the trees $T^{(k)}$ by adding an edge $r \rightarrow r^{(k)}$ for all $k = 0, \ldots, K$, and label each edge $r \rightarrow r^{(k)}$ with the subset $I_k$ of vertices of $T$. In this new tree, for all $j \in [p]$, all copies $x^{(0)}_1 \cdots x^{(0)}_j, \ldots, x^{(K)}_1 \cdots x^{(K)}_j$ of the node $x_1 \cdots x_j \in V$ are in the same stage. Recalling that $I_k$ corresponds to a unique set of stages $S_k \in S$, given two $k, k' \in \{ 0, \ldots, K \}$, if a stage $T$ of $T$ is in $S_k \cup S_{k'}$ then we insist that the nodes in stage $T$ in the copy $T^{(k)}$ and those in stage $T$ in copy $T^{(k')}$ each form a stage of their own. We call this new tree in which we have partitioned all stages according to this rule the interventional CStree for $T$ and $I$; it is denoted $T_{(T, I)}$. Every interventional CStree is also a staged tree. This is analogous to the fact that every $I$-DAG is also a DAG.

An example of an interventional CStree is depicted in Figure 3. In short, each subtree $T^{(k)}$ encodes the interventional distribution $f^{(I_k)}$ produced by targeting the stages in $S_k$ for intervention. When a stage $T$ is not targeted by either $S_k$ or $S_{k'}$ then the probabilities labeling the edges emanating from $T$ in the subtrees $T^{(k)}$ and $T^{(k')}$ have the same labeling, and hence encode the invariances of conditional probabilities stated in the definition of $M_I(T)$ above.

The tree $T_{(T, I)}$ is the representation of the model $M_I(T)$ via an interventional staged tree according to the definitions in [Duarte and Solus, 2020]. Interventional CStrees are in fact a subclass of interventional staged trees. In Appendix A.9 we give a careful proof that $M_I(T)$ is indeed an interventional staged tree model.

**Theorem 4.3.** The model $M_I(T)$ is an interventional staged tree model.
In an intervention setting, we generalize the definition of $\emptyset$ in which $M$ functional $C$-trees. Let $T$ functional $S$ of statistical equivalence of $C$-trees. We now extend these results to the family of intervention $C$-trees with respect to the targets $X$. To do so, we first extend the global Markov property for $C$-trees to the $T$. In Theorem 3.3, we established a global Markov property for $C$-trees and that the collection of targets $C$ and that the collection of targets $X$ is exactly the sequences of interventional distributions that can be generated by intervening in $T$ with respect to the targets $I$.

**Lemma 4.4.** Let $T$ be a $C$-tree and $I$ a collection of targets. Then $(f(I))_{I \in \mathcal{I}} \in \mathcal{M}_I(T)$ if and only if there exists $f(0) \in \mathcal{M}_{T}(T)$ such that $f(I)$ factorizes as in (3) with respect to $f(0)$ for all $I \in \mathcal{I}$.

### 4.3. Context-specific $I$-Markov property.

In Theorem 3.3, we established a global Markov property for $C$-trees, and in Theorem 3.5 we applied it to give a graphical characterization of statistical equivalence of $C$-trees. We now extend these results to the family of intervention $C$-trees and to the family of intervention $C$-trees. The goal is to obtain a combinatorial characterization of when $T(T,I)$ and $T(T',I')$ are statistically equivalent, i.e., when $\mathcal{M}_T(T) = \mathcal{M}_T(T')$. To do so, we first extend the global Markov property for $C$-trees to the interventional setting. We do this by generalizing the definition of $I$-DAGs introduced by Yang et al. [2018].

Let $T(T,I)$ be an intervention $C$-tree. In the following, we consider the set of contexts $C_T \cup \{\emptyset\}$ in which $\emptyset$ denotes the empty context; that is, the context in which the outcome of no variable is fixed.

Recall from Remark 3.1 that $T_{X_C=x_C} = (V_{X_C=x_C}, E_{X_C=x_C})$ denotes the context-specific subtree of $T$ for the context $X_C = x_C$ and that the collection of targets $I$ corresponds to a set of stages $S$ of $T$ via the bijection $\psi$. In the following, let $L_{(X_C=x_C,j)}$ denote the $t$-th level of $T_{X_C=x_C}$ where $\pi_t = j$ in the causal ordering $\pi$ of $T_{X_C=x_C}$. For each $X_C = x_C \in C_T \cup \{\emptyset\}$, we construct the $I$-DAG,
\[ \mathcal{G}_{X_C=x_C}, \text{ on node set } ([p] \setminus C) \cup W_T \text{ where} \]

\[ W_T := \{ w_I : I \in \mathcal{I} \setminus \{\emptyset\} \}, \]

and edges

\[ E_{X_C=x_C} \cup \{ w_I \rightarrow j : \exists x \in V_{X_C=x_C} \cap \bigcup_{T \in \psi^{-1}(I)} T \text{ such that } x \in L_{X_C=x_C}(T) \}. \]

Hence, to produce \( \mathcal{G}_{T}^{\mathcal{I}} \) from \( \mathcal{G}_{X_C=x_C} \) we add a vertex \( w_I \) to \( \mathcal{G}_{X_C=x_C} \) for each nonempty intervention target, and we connect \( w_I \) to a vertex \( j \in [p] \setminus C \) whenever such a node in \( I \) is in the level of \( T_{X_C=x_C} \) associated to the random variable \( X_j \).

We call the collection \( \mathcal{G}_{T}^{\mathcal{I}} := \{ \mathcal{G}_{X_C=x_C} \}_{X_C=x_C \in \mathcal{C}_T} \) the collection of minimal context \( \mathcal{I} \)-DAGs of \( \mathcal{T} \). For example, \( \mathcal{G}_{T}^{\mathcal{I}} \) is depicted in Figure 4 for the interventional CStree in Figure 3.

Since we include the empty context \( C = \emptyset \) in its definition, \( \mathcal{G}_{T}^{\mathcal{I}} \) always contains a context graph \( \mathcal{G}_{\emptyset} \) that records all interventions in the system at the variable level. This is because, for any \( I \in \mathcal{I} \) with \( I \neq \emptyset \), the indices in \( C = \emptyset \) of any \( y \in I \) vacuously agree with the coordinates of the empty context. Since the minimal context graphs \( \mathcal{G}_{\mathcal{T}} \) only contain contexts \( X_C = x_C \) in which there is a CSI relation, any parameter invariance under intervention outside of these contexts (where the corresponding context graph would be a complete DAG as there are no CSI relations to encode) would not be recorded in \( \mathcal{G}_{T}^{\mathcal{I}} \) without the inclusion of \( \mathcal{G}_{\emptyset} \). Including the empty context is critical for capturing such invariances.

**Example 4.1.** The interventional CStree depicted in Figure 5 has the collection \( \mathcal{G}_{T}^{\mathcal{I}} \) of interventional context graphs in Figure 6. Since the interventions all take place within the context \( X_1 = x_1^2 \), the invariance of the parameters \( f(0)(x_1) \) and \( f(1)(x_1) \) would not be represented. However, it is captured by the absence of the edge between \( w_I \) and 1 in the context graph \( \mathcal{G}_{T}^{\emptyset} \). This is justified in Proposition 4.5.

This example also illustrates how minimal contexts and interventions work together to give more refined information on the causal ordering of the variables \( X_1, X_2, X_3, X_4 \). By Theorem 3.5, any CStree \( \mathcal{T}' \) that is statistically equivalent to \( \mathcal{T} \) must encode the CSI relation \( X_4 \perp \perp X_3 \mid X_2, X_1 = x_1^2 \) with a minimal context graph \( \mathcal{G}_{X_1=x_1^2} \) that is Markov equivalent to \( \mathcal{G}_{X_1=x_1^2} \). Hence, in the causal ordering of \( \mathcal{T}' \) it must be that 1 and 2 precede either 3 or 4 (or both). However, the ordering of 2 and 3 relative to 4, for instance, can vary. As we will see in Theorem 4.6, the intervention depicted in \( \mathcal{G}_{T}^{\emptyset} \) implies that the causal ordering of any interventional CStree that is \( \mathcal{I} \)-Markov equivalent to \( \mathcal{G}_{T}^{\mathcal{I}} \) must have 4 as the last variable. This fixes the direction of the causal arrow \( 2 \rightarrow 4 \) in the context graph \( \mathcal{G}_{X_1=x_1^2} \) despite the fact that no intervention took place in this context.

In Theorem 4.6 we will see that a key feature of interventional CStrees is that the same causal information is equivalently represented by either the tree or its collection of context graphs. By comparing Figures 5 and 6, we see that it is much easier to directly read context-specific causal information from the sequence of graphs \( \mathcal{G}_{T}^{\mathcal{I}} \) as opposed to its corresponding tree representation. For example, it is immediately clear from Figure 6 that \( X_2 \) is a direct cause of \( X_4 \) in the context
Figure 5. An interventional CStree $\mathcal{T}_{(\mathcal{T}, \mathcal{I})}$ with collection of interventional targets $\mathcal{I} = \{\emptyset, \{5, 6, 11, 12, 13, 14\}\}$. 

Figure 6. The collection of minimal context $\mathcal{I}$-DAGs $\mathcal{G}_{\mathcal{I}}^T$ for the interventional CStree in Figure 5. Here $\mathcal{I} = \{\emptyset, I_1 = \{5, 6, 11, 12, 13, 14\}\}$. 

$X_1 = x_1^2$, whereas as deducing the same information from Figure 5 requires a closer analysis. This readability is a key benefit of interventional CStrees that does not hold for general interventional staged trees.

Note also that any intervention in an interventional CStree will be recorded in $\mathcal{G}_{\emptyset}^T$. This is to be expected since $f^{(\emptyset)}(X_k \mid X_{\text{pa}_G(k)}) \neq f^{(I)}(X_k \mid X_{\text{pa}_G(k)})$ holds whenever there exists some context $x = x_1 \cdots x_k$ for which $f^{(\emptyset)}(x_k \mid x_{\text{pa}_G(k)}) \neq f^{(I)}(x_k \mid x_{\text{pa}_G(k)})$. This happens whenever there is any $x \in I$ for any $I \in \mathcal{I}$, i.e., for any nonempty intervention target.

Next, we extend the global Markov property for CStrees given in Theorem 3.3. It generalizes the $\mathcal{I}$-Markov property for interventional DAG models defined in [Yang et al., 2018].

Definition 4.2 (context-specific $\mathcal{I}$-Markov Property). Let $\mathcal{T}_{(\mathcal{T}, \mathcal{I})}$ be an interventional CStree where $\emptyset \in \mathcal{I}$ and $\mathcal{T}$ has levels $(L_1, \ldots, L_p) \sim (X_1, \ldots, X_p)$. Suppose that $(f^{(I)})_{I \in \mathcal{I}}$ is a sequence of strictly positive distributions over $X_{[p]}$. We say that $(f^{(I)})_{I \in \mathcal{I}}$ satisfies the context-specific $\mathcal{I}$-Markov property with respect to $\mathcal{G}_{\mathcal{T}}^T$ if for any $X_C = x_C \in \mathcal{C}_T \cup \{\emptyset\}$:

1. $X_A \perp \perp X_B \mid X_S, X_C = x_C$ in $f^{(I)}$ for any $I \in \mathcal{I}$ and any disjoint $A, B, S \subset [p] \setminus C$ whenever $A$ and $B$ are d-separated given $S$ in $\mathcal{G}_{X_C=x_C}$, and
(2) \( f(I)(X_A \mid X_S, X_C = x_C) = f(\emptyset)(X_A \mid X_S, X_C = x_C) \) for any \( I \in \mathcal{I} \) and any disjoint \( A, S \subset [p] \setminus C \) for which \( S \cup W_T \setminus \{w_I\} \) d-separates \( A \) and \( w_I \) in \( G_{X_C=x_C}^T \).

In analogy to Theorem 4.1, we would like to observe that the set of interventional settings captured by \( \mathcal{M}_T(\mathcal{I}) \) is the same as those satisfying the context-specific \( \mathcal{I} \)-Markov property. For this to be true, we require that the intervention targets \( \mathcal{I} \) interact with (the parameters associated to) the minimal context graphs \( G_T \) in the same way that a collection of intervention targets for a DAG \( G \) interact with parameters \( f(x_k \mid X_{\text{pa}(k)}) \). That is, the intervention targets \( \mathcal{I} \) must be chosen so that all invariances defining the model \( \mathcal{M}_T(\mathcal{I}) \) are captured by Definition 4.2 (2), and vice versa. Hence, we call an intervention target \( I \) complete (with respect to a CStree \( T \)) if whenever \( x_1 \cdots x_{k-1}x_k \in I \)

(1) there exists \( X_C = x_C \in \mathcal{C}_T \) that is a subcontext of \( x_1 \cdots x_{k-1} \), and
(2) for any subcontext \( X_C = x_C \in \mathcal{C}_T \) of \( x_1 \cdots x_{k-1} \) and any \( y_1 \cdots y_{k-1} \) also having \( X_C = x_C \) as a subcontext, \( y_1 \cdots y_{k-1} \in \mathcal{I} \).

Condition (1) ensures that our intervention targets target parameters within minimal contexts, and condition (2) ensures that if a single parameter in the conditional factor \( f(X_k \mid X_{\text{pa}(k)}(k), X_C = x_C) \) is targeted, then all such parameters are targeted. This is analogous to interventions in DAG models, where targeting node \( k \) introduces a whole new conditional factor \( f(I)(X_k \mid X_{\text{pa}(k)}) \). See Figure 5 for an example of a collection of complete intervention targets. The following proposition states that we can view the context-specific \( \mathcal{I} \)-Markov property as the global Markov property for interventional CStrees when \( \mathcal{I} \) is a collection of complete intervention targets.

**Proposition 4.5.** Suppose that \( T(\mathcal{I}, \mathcal{I}) \) is an interventional CStree with \( \emptyset \in \mathcal{I} \), and \( \mathcal{I} \) a collection of complete intervention targets. Then \( (f(I))_{I \in \mathcal{I}} \) is in \( \mathcal{M}_T(\mathcal{I}) \) if and only if \( (f(I))_{I \in \mathcal{I}} \) satisfies the context-specific \( \mathcal{I} \)-Markov property with respect to \( G_T^\mathcal{I} \).

**4.4. Statistical equivalence of interventional CStrees.** Similar to the results of Verma and Pearl [1990b] and Yang et al. [2018], we can use the global Markov property for interventional CStrees (i.e., the context-specific \( \mathcal{I} \)-Markov property) to give a characterization of statistical equivalence of interventional CStrees via v-structures and skeleta of DAGs. To make this explicit, we use the following definitions.

**Definition 4.3.** Let \( T \) and \( T' \) be two CStrees with the same set of minimal contexts \( \mathcal{C} \), where \( T \) has levels \( (L_1, \ldots, L_p) \sim (X_1, \ldots, X_p) \), and \( T' \) has levels \( (L'_1, \ldots, L'_p) \sim (X_{\pi_1}, \ldots, X_{\pi_p}) \) for some jointly distributed random variables \( X_1, \ldots, X_p \) and permutation \( \pi_1 \cdots \pi_p \in \mathcal{S}_p \). If \( T \) has collection of targets \( \mathcal{I} \) and \( T' \) has collection of targets \( \mathcal{I}' \), we say that \( \mathcal{I} \) and \( \mathcal{I}' \) are compatible if there exists a bijection \( \Phi : \mathcal{I} \rightarrow \mathcal{I}' \) such that for all \( X_C = x_C \in \mathcal{C} \cup \{\emptyset\} \)

\[
\Phi(I) \rightarrow k \in G_{X_C=x_C}^{T} \iff w_\Phi(I) \rightarrow k \in G_{X_C=x_C}^{T'}.
\]

We can then extend the notion of having the same skeleton and v-structures to collections of context-graphs.

**Definition 4.4.** Let \( T, T' \) be two CStrees with targets \( \mathcal{I}, \mathcal{I}' \), respectively, and the same set of minimal contexts \( \mathcal{C} \). We say \( G_T^\mathcal{I} \) and \( G_{T'}^{\mathcal{I}'} \) have the same skeleton and v-structures if:

(1) \( \mathcal{I} \) and \( \mathcal{I}' \) are compatible,
(2) \( G_{X_C=x_C}^T \in G_T \) and \( G_{X_C=x_C}^{T'} \in G_{T'} \) have the same skeleton for all \( X_C = x_C \in \mathcal{C}_T \),
(3) \( G_{X_C=x_C}^T \in G_T \) and \( G_{X_C=x_C}^{T'} \in G_{T'} \) have the same v-structures for all \( X_C = x_C \in \mathcal{C}_T \), and
(4) \( w_I \rightarrow k \leftarrow j \) is a v-structure in \( G_{X_C=x_C}^T \) if and only if \( w_{\Phi(I)} \rightarrow k \leftarrow j \) is a v-structure in \( G_{X_C=x_C}^{T'} \) for all \( X_C = x_C \in \mathcal{C}_T \cup \{\emptyset\} \).

Definition 4.4 says that two collections of contexts graphs \( G_T^\mathcal{I} \) and \( G_{T'}^{\mathcal{I}'} \) have the same skeleton and v-structures if they have the same index set \( \mathcal{C}_T \), and the pair of graphs with a given index have the
same skeleton and v-structures up to a relabeling of the target sets. Hence, this is a generalization of two DAGs having the same skeleton and v-structures to the setting of interventional CStrees.

Example 4.2. Let \( \mathcal{T}(\mathcal{I}, \mathcal{I}') \) denote the interventional CStree depicted in Figure 7. This tree has the collection of interventional context graphs \( \mathcal{G}^T_{\mathcal{I}} \), shown in Figure 8. We compare the tree \( \mathcal{T}(\mathcal{I}, \mathcal{I}') \) with the interventional CStree \( \mathcal{T}(\mathcal{I}, \mathcal{I}) \) in Figure 5. The tree \( \mathcal{T}(\mathcal{I}, \mathcal{I}) \) has collection of targets \( \mathcal{I} = \{\emptyset, I_1 = \{5, 6, 11, 12, 13, 14\}\} \) and the tree \( \mathcal{T}(\mathcal{I'}, \mathcal{I}) \) has collection of targets \( \mathcal{I'} = \{\emptyset, I_1' = \{5, 6, 11, 12, 13, 14\}\} \). Since \( I_1 = I_1' \), we see that \( \mathcal{I} \) and \( \mathcal{I'} \) are compatible given the bijection \( \Phi \) that maps the empty set to itself and \( I_1 \) to \( I_1' \). We see then directly from Figures 6 and 8 that \( \mathcal{G}^T_{\mathcal{I}} \) and \( \mathcal{G}^{T'}_{\mathcal{I'}} \) have the same skeleton and v-structures.

The following theorem gives a characterization of statistical equivalence for interventional CStrees that extends Corollary 3.6. It is a generalization of Theorem 4.2 which was derived by Yang et al. [2018] for interventional DAG models.

**Theorem 4.6.** Let \( \mathcal{T}(\mathcal{I}, \mathcal{I}) \) and \( \mathcal{T}(\mathcal{I'}, \mathcal{I'}) \) be interventional CStrees with \( \emptyset \in \mathcal{I} \cap \mathcal{I'} \), and \( \mathcal{I} \) and \( \mathcal{I'} \) collections of complete intervention targets. Then \( \mathcal{T}(\mathcal{I}, \mathcal{I}) \) and \( \mathcal{T}(\mathcal{I'}, \mathcal{I'}) \) are statistically equivalent if and only if \( \mathcal{G}^T_{\mathcal{I}} \) and \( \mathcal{G}^{T'}_{\mathcal{I'}} \) have the same skeleton and v-structures.
5. Simulations

In the previous sections, we proved several theorems purporting CStrees as a family of models with the expressive capabilities of a general staged tree and the useful representation properties of a DAG model, such as a global Markov property yielding a Verma-Pearl-type characterization of model equivalence. We now complement these theoretical observations with an analysis of CStree model performance on simulated data. The code is available at [Duarte and Solus, 2021b].

A natural concern regarding CStrees is that restricting ourselves to modeling with a CStree when the data-generating distribution is in fact faithful to a non-CStree staged tree may result in a significant loss of predictive accuracy just for the sake of having a better representation. Empirically, this appears not to be the case. To test this we constructed, for each $p \in \{0.1, 0.2, \ldots, 0.9, 1.0\}$, 10 random staged trees on 7 binary variables with fixed causal ordering $\pi = 12 \cdots 7$. Each tree was constructed by starting with the full dependence model (every node in its own stage), then for each $k \in \{2, \ldots, 6\}$, $2^{k-1}$ (the number of possible stages in level $L_{k-1}$) Bernoulli trials $y_i \sim \text{Bernoulli}(p)$ were sampled. If $y_i = \text{true}$ then two stages in level $L_{k-1}$ were chosen uniformly at random to be combined into a single stage. This merging procedure was done iteratively over all $2^{k-1}$ Bernoulli samples before moving on to the next level $L_k$ and repeating the process. From each resulting random staged tree, 10,000 training samples were drawn and 10 validation samples.

Trees constructed with respect to a larger $p$ have less stages, and hence are the sparser models. The average number of stages of the trees constructed for value $p$ is presented in Figure 9a.

We implemented a naive algorithm, which we call BHC-CS, that learns from iid samples $D$ a CStree on $n$ variables with a known causal ordering by considering, for $k \in \{2, \ldots, n-1\}$ all possible pairwise mergings of stages in level $L_{k-1}$ – where each merging is done to (minimally) ensure the resulting tree is also a CStree – and then picks the BIC-optimal merging. When there is no longer any merging that increases BIC, the algorithm moves to level $L_k$ and repeats the process. By Proposition 3.8, this is a consistent algorithm for learning CStrees with a known causal ordering.

Our algorithm is the CStree equivalent of the backwards hill-climbing algorithm for learning staged trees [Carli et al., 2020], which we denote BHC-S.

To relate the performance of a learned CStree to a learned staged tree, we let the BHC-CS and BHC-S attempt to learn each true (i.e., data-generating) staged tree based on the training data. For each $p$, we report the average structural Hamming distance between the true staged tree and the learned trees in Figure 9b. As expected, the trees learned by BHC-S are closer to the true.
trees than the learned CStrees, particularly for moderately sparse trees where the true tree can assume many more staging patterns than sparse CStrees. This is because, in most cases, the data-generating tree will not be a CStree, and hence the best we can hope for is a decent approximation of the true tree.

While the learned CStrees are only approximations of the true staged tree, we find that they function equally as well as the learned staged trees. To see this, we performed a posterior predictive check using the ten validation samples. For each value of $p$ and each variable $X_i$, the learned trees were tasked with predicting the value of $X_i$ in each validation sample given the values of all remaining variables. The average predictive accuracy (i.e., the average proportion of times each tree correctly predicted the missing value) is recorded in Figure 9c. We see that, despite learned CStrees only being able to approximate the true staged tree, their average predictive accuracy is essentially the same as that of the learned staged trees. This suggests that one can safely use CStrees without sacrificing predictive accuracy while gaining the representation results for CStrees proven in the previous sections.

According to Figure 9b, despite being consistent, the backwards hill-climbing algorithms appear to rarely learn the true tree. This is attributable to the sample size required to exactly learn the stages in the higher levels of the trees. To see the consistency of BHC-CS in practice, we can increase the sample size and observe a decrease in structural Hamming distance as depicted in Figures 10b and 10c for sample sizes 1,000, 10,000 and 100,000 drawn from random binary CStrees on 6 binary variables. The random CStrees are constructed analogously to the random staged trees for the previous experiment. However, the merging of stages in each level so as to always produce a CStree can quickly lead to much sparser models. So the number of Bernoulli trials (for level $L_{k-1}$ and merging probability $p$) was adjusted to be $\left\lfloor \frac{2^{k-1}}{1 + 4k(p - p^2)} \right\rfloor$. This adjustment yields an approximately linear expected number of stages, as depicted in Figure 10a.

Figure 10b demonstrates that increasing the sample size strictly improves our estimates of the true CStree. Figure 10c, which compares the number of stages of each of the true CStrees with its structural Hamming distance from the learned CStree, suggests that for at least 10,000 samples, the learned CStree is quite close to the true CStree when the true tree has at most 1/3 of the number of possible stages for a binary CStree on 6 variables – 64. For the same sparsity level, we see that 100,000 samples appears sufficient to learn almost all of the CStrees exactly.
6. Applications to Real Data

We apply the results on CSTrees and interventional CSTrees to two real data sets, demonstrating how the alternative representation of a (interventional) CSTree via its context graphs allows us to quickly read context-specific causal information inferred from data. The first data set is purely observational and it studies risk factors in coronary heart disease; it is available with the R [R Core Team, 2020] package bnlearn [Scutari, 2010]. The second data set is a mixture of observational and interventional data on the regulation of the expression of proteins critical to learning in mice; it is available at the UCI Machine Learning Repository [Dua and Graff, 2017]. The code is available at [Duarte and Solus, 2021b].

6.1. Coronary heart disease data. The data set coronary, included in the bnlearn package, contains samples from 1814 men measuring probable risk factors for coronary heart disease with binary variables. The list of variables with their outcome set is: $S$: smoking {"no","yes"}, $MW$: strenuous mental work {"no","yes"}, $PW$: strenuous physical work {"no","yes"}, $P$: systolic blood pressure, $\{ (>140), (<140) \}$, $L$: ratio of $\beta$ to $\alpha$-lipoproteins $\{ (<3), (>3) \}$, and $F$: family history of coronary heart disease {"negative","positive"}. For the CSTree depicted in Figure 11, an arrow pointing downwards indicates the first outcome of the variable at that level, and one pointing upwards indicates the second outcome.

To learn the BIC-optimal CSTree given data on six binary variables, we would have to score $1.1362 \times 10^{22}$ models, which is on the order of the number of DAGs (or MECs of DAGs) on 11 variables. To limit the exponential blow-up in the number of computations needed, we employ the BHC-CS algorithm introduced in Section 5. Since BHC-CS assumes a known causal ordering of the tree, we run BHC-CS one time for each of the $6! = 720$ causal orderings and then select the BIC-optimal output over all of these runs, an adaptation we refer to as BHC-CS-perm. Since the sample size is roughly 2,000, assuming the underlying CSTree structure is sparse, we can expect a decent approximation of the data-generating structure based on the simulations in Figure 10. The learned CSTree is depicted in Figure 11a.

The tree depicted in Figure 11a has a total of 15 stages, which is less than a third of the possible 64 stages for a binary CSTree on 6 variables. There are 7 non-singleton stages which, respectively, have the following associated CSI relations via Lemma 3.1:

$$PW \perp P \mid MW = yes, S \perp MW, PW \mid P = ( > 140 ), S \perp MW \mid P = ( < 140 ), PW = yes,$$

$$L \perp MW, PW, P \mid S = no, L \perp MW, PW \mid S = yes, P = ( > 140 ),$$

$$L \perp MW \mid S = yes, PW = yes, P = ( < 140 ), \text{ and } F \perp S, MW, PW, P, L.$$  

The context for each of the above CSI relations is the stage-defining context for its associated stage. By taking the context-specific closure of these seven CSI relations, we recover the minimal context graphs $G_{\text{coronary}}$ depicted in Figure 11b.

The empty context for $T_{\text{coronary}}$ is the complete DAG on $MW, PW, P, S$ and $L$ with $F$ independent of all other variables. DAG learning algorithms such as those in the bnlearn package in R learn DAGs for this data set with similar structure: $F$ adjacent to only one variable and only two non-adjacencies among the remaining variables. However, the CSTree structure reveals context-specific causal information that is overlooked by such DAG models. For instance, the context graph $G_{MW=\text{yes}}$ encodes two v-structures, purporting physically strenuous work ($PW$) and systolic blood pressure ($P$) as independent predictors of both smoking ($S$) and the ratio of $\beta$ to $\alpha$-lipoproteins ($L$) in the context that the individual has mentally strenuous work ($MW = yes$). By Corollary 3.6, any CSTree statistically equivalent to $T_{\text{coronary}}$ must have the same minimal contexts and same skeletons and v-structures in each of its minimal context graphs. Hence, the presence of these two v-structures also fixes the directions of these arrows in the empty context graph $G_{\emptyset}$, thereby providing more refined causal information than carried by the lone DAG $G_{\emptyset}$. The representation theorem (Theorem 3.3) and the resulting characterization of statistical equivalence (Corollary 3.6)
allow us to easily see these causal relations directly from the context graphs, without any further analysis that may be required to deduce the same results by working only with the staged tree representation $T_{\text{coronary}}$. In a similar fashion, one can quickly read off context-specific conditional independence relations from the context-graphs that may be difficult to deduce directly from the staged tree representation. For example, we see directly from the context graph $G_{P=(>140)}$ that physically strenuous work and mentally strenuous work are independent from smoking and the ratio of $\beta$ to $\alpha$-lipoproteins in the context that systolic blood pressure is high ($P=(>140)$). In general, reading CSI relations $X_A \perp \perp X_B \mid X_S, X_C = x_C$ where $A$ and $B$ are not singletons from a staged tree can be a challenging task, but it becomes straightforward by restricting to CStrees and utilizing their representation via minimal context-graphs.

6.2. Mice protein expression data. The data set available at the UCI Machine Learning Repository [Dua and Graff, 2017] records expression levels of 77 different proteins/protein modifications measured in the cerebral cortex of mice. Each mouse is either a control or a Ts65Dn trisomic Down Syndrome mouse. Each mouse was either injected with saline or treated with the drug memantine, which is believed to affect associative learning in mice. The mice were then trained in context fear conditioning (CFC), a task used to assess associative learning [Radulovic et al., 1998]. The standard CFC protocol divides mice into two groups: the context-shock (CS) group, which is placed into a novel cage, allowed to explore, and then receives a brief electric shock, and the shock-context (SC) group, which is placed in the novel cage, immediately given the electric shock, and thereafter is allowed to explore. The expression levels of 77 different proteins were measured from eight different classes of mice, defined by whether the mouse is control (c) or trisomic (t), received memantine (m) or saline (s), and whether it was in a CS or SC group for the learning task. The eight classes are denoted as c-CS-s, c-CS-m, c-SC-s, c-SC-m, t-CS-s, t-CS-m, t-SC-s, t-SC-m. There are 9, 10, 9, 10, 7, 9, 9, and 9 mice in each class, respectively. Fifteen measurements
of each protein were registered per mouse, yielding a total of 1080 measurements per protein. Each measurement is regarded as an independent sample.

In the following, we treat the 135 measurements taken from the group c-SC-s as observational data and the 150 measurements taken from the group c-SC-m as interventional data, taking treatment with memantine as our intervention. We consider the expression levels of four proteins, each of which is believed to discriminate between between the classes c-SC-s and c-SC-m (see [Higuera et al., 2015, Table 3, Column 2]). As our models are for discrete random variables only, we discretized the data set using the quantile method. The result is four binary random variables, one for each protein considered, with outcomes “high (expression level)” and “low (expression level).” Hence, our goal is to learn an optimal interventional CStree on four binary variables given the observational data and one interventional data set. Here, we assume that the intervention targets are unknown since we cannot be certain which proteins, at which expression levels (i.e. in which contexts), are directly targeted by the drug memantine.

There are 59,136 CStrees on four binary random variables. As the intervention targets are latent, we need to consider all possible interventions in a given observational CStree. For a CStree on four binary variables, the number of models to be scored can be as large as $2^{15}$. To avoid the time complexity issues of testing many models for each of 59136 trees, we instead first learn the BIC-optimal equivalence class of CStrees for the data and then score all possible interventional CStrees that arise by targeting any subset of the stages in any element of this equivalence class.

6.2.1. pCAMKII, pPKCG, NR1 and pS6. A representative of the BIC-optimal equivalence class for the observational data with the proteins pCAMKII, pPKCG, NR1 and pS6 is given in Figure 12; its context graphs are depicted in Figure 13. According to Corollary 3.6, the CStree in Figure 12 is in an equivalence class of size two, where the other element is given by swapping pCAMKII and pS6 in the causal ordering. This corresponds to reversing the arrow between these two nodes in the graph $G_{pPKCG=\text{low}}$. Notice also that the arrow pPKCG→NR1 is covered in the DAG $G_{\emptyset}$, and hence reversing this arrow would result in a Markov equivalent DAG. However, according to Corollary 3.6, this arrow is fixed among all elements of the equivalence class due to the v-structure in the context graph $G_{pCAMKII=\text{high}}$.

We can try to use our interventional data drawn from the class c-SC-m to distinguish between the two elements of this equivalence class, and hence determine the complete causal ordering of the variables. We compute the BIC-optimal interventional CStree that arises from any choice.
of nonempty complete interventions targets in either of the two elements of the observational equivalence class. The resulting BIC-optimal interventional CStree is depicted in Figure 14, and its interventional contexts graphs are given in Figure 15. Comparing these two figures, we already see how the representation of an interventional CStree via its context graphs is more compact and easier to read than its staged tree representation. We also see that precisely one stage is targeted for intervention and it is in the context $p_{CAMKII} = \text{high}$. Hence, the interventional context graphs contain two arrows pointing from an interventional node $w_{\text{mem}}$ (representing intervention via memantine) into the node $p_{PKCG}$, one arrow in the $G_{\emptyset}$ and the other in $G_{p_{CAMKII} = \text{high}}$. Note that no arrow is added to $G_{p_{CAMKII} = \text{low}}$, as no stages in this context were targeted by the intervention. While this intervention introduces new v-structures, none of the new v-structures fix
edges that were not already fixed in the observational context graphs. Hence, a targeted intervention at pCAMKII or pS6 in the context that pPKCG = low is needed to distinguish the true causal structure among the proteins.

6.2.2. pPKCG, pNUMB, pNR1 and pCAMKII. To illustrate the refinement of equivalence classes via intervention, we can consider another set of four proteins: pPKCG, pNUMB, pNR1 and pCAMKII. A representative of the BIC-optimal equivalence class given the observational data for these four proteins is depicted in Figure 16, and its representation as a sequence of context graphs is in Figure 17. Via Corollary 3.6, one can deduce that the equivalence class of the CStree in Figure 16 contains three additional CStrees, which are shown in Figure 20 in Appendix B.

We compute the BIC of every interventional CStree that arises for a set of valid intervention targets in one of the CStrees in the equivalence class of the tree in Figure 16. Given the (finite) data, the highest BIC score is achieved four times by the trees making up two different equivalence classes of interventional CStrees, each having only complete intervention targets. The interventional context graphs of these two classes are depicted in Figure 18, and the four trees appear in Figures 21 and 22 in Appendix B. By Theorem 4.2, we see directly from Figure 18a that the BIC-optimal interventional CStree with these contexts graphs is in an equivalence class of size one, whereas there are three trees in the equivalence class represented by Figure 18b. As the BIC score cannot distinguish between these two equivalence classes given the small sample size we performed a bootstrap, producing 1000 replicates and computing the BIC score for both equivalence classes for each replicate. The mean of the bootstrapped BICs is slightly higher for the equivalence class of size one. Hence, we take the context graphs in Figure 18a to be the context-specific causal structure of the data-generating interventional setting.
7. Discussion

Among the features of DAG models that make them so successful in practice is their admittance of a global Markov property, which provides a combinatorial criterion describing the conditional independence relations satisfied by any distribution that factorizes according to the given DAG. The global Markov property for DAGs leads to the many combinatorial characterizations of model equivalence [Andersson and Perlman, 1997, Chickering, 1995, Verma and Pearl, 1990b] that drive some of the most popular causal discovery algorithms to date [Chickering, 2002, Solus et al., 2021, Spirtes et al., 2000].

From this perspective, the results established in this paper purport CStrees as the natural extension of DAG models to a family capable of expressing diverse context-specific conditional independence relations while maintaining the intuitive representation of a DAG. CStrees admit a global Markov property generalizing that of DAGs (Theorem 3.3), which leads to a characterization of model equivalence that generalizes a classical characterization of DAG model equivalence (Corollary 3.6). Moreover, these results generalize to interventional models (Section 4), making CStrees the first family of context-specific interventional models admitting a characterization of model equivalence. As we saw in real data examples (e.g., Figure 11), the representation of a CStree via its sequence of contexts graphs offers the user a quick way to read context-specific causal information, whereas reading the same information from the staged tree or CEG representation requires a more careful analysis. The simulations in Section 5 further suggest that one can safely model with a CStree instead of a more general staged tree without sacrificing predictive accuracy. Moreover, the BIC is a locally consistent scoring criterion for CStrees (Proposition 3.8), setting the stage for the development of consistent CStree learning algorithms that are more efficient than the naive BHC-CS-perm algorithm implemented in Subsection 6.1.

Natural directions for future research include a context-specific PC algorithm for learning CStrees and a similar extension of the Greedy Equivalence Search (GES) [Chickering, 2002]. The observation that BIC is locally consistent for CStrees suggests that Meek’s Conjecture [Chickering, 2002, Theorem 4] could be extended to this family of models. The development of a consistent extension of GES to CStrees would provide a faster alternative to BHC-CS-perm, which necessarily considers all possible causal orderings of the tree (similar to the greedy permutation search (GSP) [Raskutti and Uhler, 2018] for DAGs). The results of Section 4 also suggest that the recent causal discovery algorithms that incorporate a mixture of observational and interventional data [Hauser and Bühlmann, 2012, Kocaoglu et al., 2019, Wang et al., 2017, Yang et al., 2018] can likely be extended to the context-specific setting via CStrees. Finally, it would also be interesting to see extensions of CStree models to the non-discrete data and/or causally insufficient settings.
ACKNOWLEDGEMENTS

Eliana Duarte was supported by the Deutsche Forschungsgemeinschaft DFG under grant 314838170, GRK 2297 MathCoRe, by the FCT grant 2020.01933.CEECIND, and partially supported by CMUP under the FCT grant UIDB/00144/2020. Liam Solus was supported a Starting Grant (No. 2019-05195) from Vetenskapsrådet (The Swedish Research Council), the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, and the Göran Gustafsson Stiftelse. The authors would also like to thank Danai Deligeorgaki, Christiane Görgen, Manuele Leonelli, and Gherardo Varando for helpful discussions.

APPENDIX A. PROOFS OMITTED FROM THE MAIN TEXT

A.1. Proof of Lemma 3.1. To prove the statement, we need to show that

\[ f(x_k \mid x_{[k-1]\setminus C_i}, x_{C_i}) = f(x_k \mid x_{C_i})\]

for any \( f(x_k, x_{[k-1]\setminus C_i}, x_{C_i}) > 0 \), where \( C_i = \bigcap_{v \in S_i} v \) and \( x_{C_i} := y_{C_i} \) for any \( y \in S_i \). Since \( T \in T_{CS} \) then it is uniform, stratified and compatibly-labeled. Hence, it follows for any \( v, w \in S_i \) with \( v = x := x_1 \cdots x_{k-1} \) and \( w = x' := x'_1 \cdots x'_{k-1} \)

\[ f(X_k \mid x_1 \cdots x_{k-1}) = f(X_k \mid x'_1 \cdots x'_{k-1}).\]

Furthermore, since \( T \in T_{CS} \), then there exists some \( C \subset [k] \) and \( x_C \in R_C \) such that \( S_i \) is exactly the collection of vertices \( y \) in \( L_k \) satisfying \( y_C = x_C \). Hence, \( C = C_i \). It follows that

\[ f(X_k \mid x_{C_i}, x_{[k-1]\setminus C_i}) = f(X_k \mid x_{C_i}, x'_{[k-1]\setminus C_i}).\]

Since \( T \in T_{CS} \), it follows that this equality holds for any \( x'_{[k-1]\setminus C_i} \in R_{[k-1]\setminus C_i} \). This fact is equivalent to \( X_k \perp X_{[k-1]\setminus C_i} \mid x_{C_i} = x_{C_i} \).

A.2. Proof of Lemma 3.2. Suppose that \( X_C = x_C \notin C_T \). Then, by definition of a minimal context, there must exist \( \emptyset \neq T \subseteq C \) such that

\[ X_A \perp X_B \mid X_{S \setminus T}, X_{C \setminus T} = x_{C \setminus T} \in J(T).\]

Picking \( T \) to be any maximal subset of \( C \) with respect to this property yields \( X_{C \setminus T} = x_{C \setminus T} \in C_T \), and hence \( X_C = x_C \) has the minimal context \( X_{C \setminus T} = x_{C \setminus T} \) as a subcontext.

By Lemma 3.1, for any stage \( S_i \subset L_{k-1} \)

\[ X_k \perp X_{[k-1]\setminus C_i} \mid x_{C_i} = x_{C_i} \in J(T),\]

and by definition those statements generate \( J(T) \). Notice that the only two context-specific conditional independence axioms that change the context \( X_C = x_C \) are specialization and absorption. Since specialization can never produce a minimal context, then they can only be produced by an application of the absorption axiom. However, any statement \( X_A \perp X_B \mid X_{S \setminus T}, X_{C \setminus T} = x_{C \setminus T} \) produced by applying absorption to a statement \( X_A \perp X_B \mid X_S, X_C = x_C \) will have context \( X_{C \setminus T} = x_{C \setminus T} \) as a subcontext of \( X_C = x_C \). Hence, each minimal context in \( C_T \) must be a subcontext of (at least) one \( x_{C_i} = x_{C_i} \), the contexts of the generators of \( J(T) \).

A.3. Proof of Theorem 3.3. We first show that (1) and (2) are equivalent. Suppose that \( f \in M_{(T, \theta)} \). Then \( f \) entails all context-specific conditional independence relations

\[ X_k \perp X_{[k-1]\setminus C_i} \mid x_{C_i} = x_{C_i}\]

from Lemma 3.1. Hence, \( f \) is Markov to \( J(T) \), and therefore entails all context-specific conditional independence relations in \( J_{X_C = x_C} \), for each \( X_C = x_C \in C_T \).

Conversely, suppose that \( f \in M(G_T) \). Then for all \( X_C = x_C \in C_T \), \( f \) entails \( X_A \perp X_B \mid X_S, X_C = x_C \) whenever \( A \) and \( B \) are d-separated given \( S \) in the minimal I-MAP \( G_{X_C = x_C} \) of
\[ \mathcal{J}_{X_C=x_C}. \] To see that \( f \) factorizes according to \( \mathcal{T} \), it suffices to show that, for all \( k \in [p-1] \), and all stages \( S_i \) in level \( L_{k-1} \), \( f \) entails

\[ X_k \perp X_{[k-1] \setminus C_i} \mid X_{C_i} = x_{C_i}, \]  

(4)

where \( X_{C_i} = x_{C_i} \) is the context associated to the stage \( S_i \) in Lemma 3.1. By definition of \( \mathcal{J}(\mathcal{T}) \), each CSI relation (4) is in \( \mathcal{J}(\mathcal{T}) \). So by Lemma 3.2, either \( X_{C_i} = x_{C_i} \in \mathcal{C}_\mathcal{T} \) or there exists \( C \subseteq C_i \) such that \( X_C = x_C \in \mathcal{C}_\mathcal{T} \) and (4) is implied by the specialization of

\[ X_k \perp X_{[k-1] \setminus C_i} \mid X_{C_i \setminus C}, X_C = x_C \in \mathcal{J}_{X_C=x_C}. \]  

(5)

By weak union, it follows that

\[ X_k \perp X_j \mid X_{[k-1] \setminus (C_i \cup \{j\})}, X_{C_i \setminus C}, X_C = x_C \in \mathcal{J}_{X_C=x_C} \]

for all \( j \in [k-1] \setminus C_i \), and hence, the minimal I-MAP \( \mathcal{G}_{X_C=x_C} \) does not contain the edges \( j \rightarrow k \) for all \( j \in [k-1] \setminus C_i \). Since independence models for DAGs are compositional (see [Sadeghi and Lauritzen, 2014]), it follows that \( k \) and \( [k-1] \setminus C_i \) are d-separated given \( C_i \setminus C \) in \( \mathcal{G}_{X_C=x_C} \). Hence, \( f \) entails the statements in (5). Applying specialization, it follows that \( f \) entails the CSI relations in (4), and therefore factorizes according to \( \mathcal{T} \).

We now show that (2) and (3) are equivalent. Let \( X_C = x_C \in \mathcal{C}_\mathcal{T} \), and set \( g(X_{[p]\setminus C}) := f(X_{[p]\setminus C} \mid X_C = x_C) \). Since \( f \) is Markov to \( \mathcal{G}_\mathcal{T} \), whenever \( A \) and \( B \) are d-separated in \( \mathcal{G}_{X_C=x_C} \) given \( S \), we have that \( f \) entails \( X_A \perp X_B \mid X_S, X_C = x_C \); or equivalently,

\[ \frac{f(x_A, x_B, x_S, x_C)}{f(x_B, x_S, x_C)} = \frac{f(x_A, x_S, x_C)}{f(x_S, x_C)}, \]

for any \((x_A, x_B, x_S) \in \mathcal{R}_{AUBJS}\). Since for any \( x_{[p]\setminus C} \in \mathcal{R}_{[p]\setminus C}\),

\[ g(x_{[p]\setminus C}) = \frac{1}{f(x_C)} f(x_{[p]\setminus C}, x_C), \]

then

\[ g(x_A, x_B, x_S) = \sum_{y \in \mathcal{R}_{[p]\setminus AUBJS}} \frac{f(y, x_A, x_B, x_S, x_C)}{f(x_C)} = \frac{1}{f(x_C)} f(x_A, x_B, x_S, x_C), \]

and similarly for \( g(x_B, x_S), g(x_A, x_S), \) and \( g(x_S) \). Hence, whenever \( A \) and \( B \) are d-separated in \( \mathcal{G}_{X_C=x_C} \) given \( S \), we have that

\[ \frac{g(x_A, x_B, x_S)}{g(x_B, x_S)} = \frac{g(x_A, x_S)}{g(x_S)}. \]

Therefore, \( g \) entails \( X_A \perp X_B \mid X_S \) whenever \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \) for all \( X_C = x_C \in \mathcal{C}_\mathcal{T} \) if and only if \( f \) is Markov to \( \mathcal{G}_\mathcal{T} \). It follows that \( f \) is Markov to \( \mathcal{G}_\mathcal{T} \) if and only if for all \( X_C = x_C \in \mathcal{C}_\mathcal{T} \),

\[ f(X_{[p]\setminus C} \mid X_C = x_C) = g(X_{[p]\setminus C}) = \prod_{k \in [p] \setminus C} \frac{g(X_k \mid X_{pa}X_C=x_C(k))}{g(X_{pa}X_C=x_C(k))}, \]

\[ = \prod_{k \in [p] \setminus C} \frac{f(X_k, X_{pa}X_C=x_C(k), X_C = x_C)}{f(X_{pa}X_C=x_C(k), X_C = x_C)}, \]

\[ = \prod_{k \in [p] \setminus C} f(X_k \mid X_{pa}X_C=x_C(k), X_C = x_C), \]

which completes the proof.
A.4. Proof of Lemma 3.4. Suppose, for the sake of contradiction, that there exists \( X_C = x_C \in C_T \) such that \( X_C = x_C \notin C_T' \). Then, by definition of the set \( C_T \) and Lemma 3.2, there must exist a CSI relation \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) in \( J(T) \) that is not implied by specialization of any statement \( X_A \perp \perp X_B \mid X_S \cup J \setminus T, X_C \setminus T = x_C \setminus T \). Since \( X_C = x_C \notin C_T' \), it follows that either \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) is not a CSI relation in \( J(T') \) or there exists some subcontext \( X_C \setminus T = x_C \setminus T \in C_T' \) such that the statement \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) is implied by the statement \( X_A \perp \perp X_B \mid X_S \cup J \setminus T, X_C \setminus T = x_C \setminus T \) encoded by \( g'_{X_C \setminus T = x_C \setminus T} \in G_T \). It follows from Theorem 3.3 that the statement \( X_A \perp \perp X_B \mid X_S \cup J \setminus T, X_C \setminus T = x_C \setminus T \) is in \( J(T') \). However, such a statement cannot be in \( J(T) \), as this would imply that the statement \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) is obtained by specialization from a statement of the form \( X_A \perp \perp X_B \mid X_S \cup J \setminus T, X_C \setminus T = x_C \setminus T \) in \( J(T) \). This latter fact would contradict our initial assumption that \( X_C = x_C \) is a minimal context. Hence, we may assume that there is a minimal context \( X_C = x_C \), such that the CSI relation \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) is in \( J(T) \) but not in \( J(T') \).

We now note that, by Duarte and Görgen [2020], the model \( M_T \) is equal to an irreducible algebraic variety intersected with the probability simplex. That is, \( M_T = V(P_T) \cap \Delta_{|R|-1}^o \) where \( P_T \) is a prime ideal in a polynomial ring and \( V(P_T) \) is the set of all points in \( \mathbb{C}^{|R|} \) that vanish on the polynomials in \( P_T \). The same holds for \( T' \), \( M_{T'} = V(P_{T'}) \cap \Delta_{|R|-1}^o \). Since \( M_T = M_{T'} \), it follows that their closures with respect to the Zariski topology are equal, namely \( V(P_T) = V(P_{T'}) \) and hence \( P_T = P_{T'} \). In particular, every equation that is satisfied by every distribution in \( M_T \) is also an equation satisfied by every distribution in \( M_{T'} \). Since \( X_A \perp \perp X_B \mid X_S, X_C = x_C \) is in \( J(T) \), then every distribution in \( M_T \) satisfies \( X_A \perp \perp X_B \mid X_S, X_C = x_C \). By [Sullivant, 2018, Proposition 4.1.6] restricted to the context-specific setting, there is a set of polynomials associated to the statement \( X_A \perp \perp X_B \mid X_S, X_C = x_C \), which we denote by \( I_{A \perp \perp B \mid S, X_C = x_C} \), that vanish at every distribution in \( M_T \). In particular, \( I_{A \perp \perp B \mid S, X_C = x_C} \subset P_T = P_{T'} \). Hence every polynomial in \( I_{A \perp \perp B \mid S, X_C = x_C} \) vanishes at every distribution in \( M_{T'} \), which means that every distribution in \( M_T \) satisfies the statement \( X_A \perp \perp X_B \mid X_S, X_C = x_C \). But this implies \( X_A \perp \perp X_B \mid X_S, X_C = x_C \in J(T') \), a contradiction. Hence, \( M_T \) and \( M_{T'} \) have the same set of minimal contexts.

A.5. Proof of Theorem 3.5. Suppose \( T \) and \( T' \) are statistically equivalent. By Lemma 3.4, it follows that \( C := C_T = C_{T'} \). So we need to show that for all \( X_C = x_C \in C \), for any disjoint subsets \( A, B, S \subset [p] \setminus C \) with \( A, B \neq \emptyset \), that \( A \) and \( B \) are \( d \)-separated given \( S \) in \( G_{X_C = x_C} \), if and only if \( A \) and \( B \) are \( d \)-separated given \( S \) in \( G'_{X_C = x_C} \). For the sake of contradiction, suppose that \( A \) and \( B \) are \( d \)-separated given \( S \) in \( G_{X_C = x_C} \) but \( d \)-connected given \( S \) in \( G'_{X_C = x_C} \). Let \( \pi = <i_0, \ldots, i_M> \) be a \( d \)-connecting path between \( i_0 \in A \) and \( i_M \in B \) given \( S \) in \( G'_{X_C = x_C} \). Let \( \mathcal{G} \subset G'_{X_C = x_C} \) be the subgraph of \( G'_{X_C = x_C} \) consisting of all nodes and edges on \( \pi \) together with all nodes and edges on a directed path from any node \( i_0 \) of \( \pi \) that is the center of a collider subpath of \( \pi \) to a node in \( S \). Suppose also that any remaining nodes of \( S \) not captured in the above paths are included in \( G' \) as isolated nodes. Let \( V \) denote the set of nodes of \( G' \). By [Meek, 2013, Lemma 12], there exists a discrete distribution \( P_V \sim X_V \) that is Markov to \( G' \) for which \( X_{i_0} \not\perp \perp X_{i_M} \mid X_S \) holds in \( P_V \). As \( G' \) is a subDAG of \( G'_{X_C = x_C} \), it follows that the subword of the causal ordering of \( T' \) on the elements of \( V \) is a linear extension of \( G' \). Hence, we can factor \( P_V \) according to a subtree of \( T' \) in the following way:

Let \( f_V(x) = \prod_{i \in V} f_V(x_i \mid x_{pa_{G'}(i)}) \) be the probability mass function for \( P_V \). For each \( i \in V \), consider its associated level (without loss of generality, level \( L_{i-1} \)) in the tree \( T' \). Let \( x = (x_1, \ldots, x_p) \in R \) be any outcome of \( (X_1, \ldots, X_p) \) that includes the context \( X_C = x_C \). For every \( i \in V \), it follows that the root-to-leaf path in \( T' \) corresponding to the outcome \( x \) passes through exactly one stage in level \( L_{i-1} \). We assign the parameters on the edges emanating from nodes in this the stage value \( f_V(x_i \mid x_{pa_{G'}(i)}) \), for all \( x_i \in R_{\{i\}} \). We then randomly generate a sequence of
there exist a collection of measures \( \alpha_1^{(i)}, \ldots, \alpha_{|R(i)|}^{(i)} \in (0, 1) \) that sum to one and assign these values to all other parameters on any edge emanating from level \( L_{i-1} \), one parameter to each edge of every floret (always assigned in the same order for every floret, say top-to-bottom). In particular, we let \( \alpha_{x_i}^{(i)} \) be the element in this sequence that is always assigned to the edge emanating from the floret that corresponds to the outcome \( x_i \) of \( X_i \). We similarly assign parameters to all edges on levels of \( T' \) corresponding to variables \( X_i \) with \( i \notin V \). It follows that such a specification of parameters factors according to \( T' \), and hence specifies a distribution \( \mathbb{P} \in \mathcal{M}_{T'} \) with density \( f \). (This is because the assignment of parameters we have made corresponds to a staging of a tree with the same causal ordering as \( T' \) whose stages are a coarsening of those in \( T' \).) Moreover, for every \( x_V \in R_V \) we have that \( f(x_V | x_C) = f_V(x_V) \).

It follows that \( X_{i_0} \not\perp X_{i_M} | X_S, X_C = x_C \) holds in \( \mathbb{P} \). As \( \mathcal{M}_{T} = \mathcal{M}_{T'} \), and \( f \) factors according to \( T' \), it must be that \( f \) also factors according to \( T \). By Theorem 3.3, we know that \( f \in \mathcal{M}(\mathcal{G}_{T}) \). Hence, as \( X_C = x_C \in C_T \), and \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \), then it must be that \( i_0 \) and \( i_M \) are also d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \). Hence, as \( f \in \mathcal{M}(\mathcal{G}_{T}) \), it must be that \( \mathbb{P} \) entails \( X_{i_0} \not\perp X_{i_M} | X_S, X_C = x_C \), which is a contradiction. Thus, we conclude that \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \) if and only if \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}'_{X_C=x_C} \). Hence, \( \mathcal{G}_{X_C=x_C} \) and \( \mathcal{G}'_{X_C=x_C} \) are Markov equivalent for all \( X_C = x_C \in C \).

Suppose now that \( \mathcal{G}_{X_C=x_C} \) and \( \mathcal{G}'_{X_C=x_C} \) are Markov equivalent for all \( X_C = x_C \in C \). We need to show that \( \mathcal{M}_{T} = \mathcal{M}_{T'} \). If \( f \in \mathcal{M}_{T} \), by Theorem 3.3, \( f \) is Markov to \( \mathcal{G}_{T} \) and hence entails \( X_A \perp X_B | X_S, X_C = x_C \) whenever \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \). As \( \mathcal{G}_{X_C=x_C} \) and \( \mathcal{G}'_{X_C=x_C} \) are Markov equivalent, it follows that \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}'_{X_C=x_C} \) if and only if \( A \) and \( B \) are d-separated given \( S \) in \( \mathcal{G}_{X_C=x_C} \). Hence, \( f \) is also Markov to \( \mathcal{G}_{T'} \). Again by Theorem 3.3, it follows that \( f \in \mathcal{M}_{T'} \). By symmetry of this argument it follows that \( \mathcal{M}_{T} = \mathcal{M}_{T'} \), which completes the proof.

### A.6. Proof of Proposition 3.7.

Following the formula from [Duarte et al., 2021, Proposition 11], we find the maximum likelihood estimates for the parameters of the model. Let \( \hat{y}_{\theta(e)} \) be a parameter associated to the edge \( e = x_{[k-1]} \to x_{[k]} \) in \( T \). The quotient \( u_{X, [k]} / u_{X, [k-1]} \) is the empirical estimate for the transition probability from \( x_{[k-1]} \) to \( x_{[k]} \). By [Duarte et al., 2021, Remark 12] to obtain the maximum likelihood estimate for \( \hat{y}_{\theta(e)} \) we consider all fractions \( u_{X', [k]} / u_{X', [k-1]} \) such that \( \theta(x'_{[k-1]} \to x'_{[k]}) = \theta(e) \) and aggregate them by adding their numerators and denominators separately. The complete set of those fractions is indexed by the elements in the stage \( S_j \subset L_{k-1} \) that contains the node \( x_{[k-1]} \). Since \( T \) is a CSTree, any two vertices \( x', x'' \) in the same stage \( S_j \) are in the same level and satisfy \( x'_{C_j} = x''_{C_j} \) for some context \( C_j \) associated to \( S_j \) (see Lemma 3.1). Therefore

\[
\hat{y}_{\theta(e)} = \frac{\sum_{x'_{[k]} \in C_j} u_{x', [k]} / u_{x', [k-1]}}{\sum_{x''_{[k]} \in C_j} u_{x'', [k]} / u_{x'', [k-1]}} = \frac{u_{x, C_j \cup \{k\}}}{u_{x, C_j}}.
\]

Finally, \( \hat{p}_x = \prod_{k=1}^{L_k} \hat{y}_{\theta([x_{[k-1]}] \to x_{[k]})} \) and using the formula for \( \hat{y}_{\theta(x_{[k-1]} \to x_{[k]})} \) yields the desired equation (3.7). The last assertion in the proposition follows from [Duarte and Solus, 2020, Example 3.2].

### A.7. Proof of Proposition 3.8.

The proof of this claim is analogous to the proof of [Chickering, 2002, Lemma 7]. Essentially, BIC is a decomposable score function for a CSTree in the sense that there exist a collection of measures \( s(S_i, x_k^j) \) that depend only on the data \( D \) restricted to the outcomes passing through stage \( S_i \subset L_{k-1} \) and their values \( x_k^j \in [d_k] \) such that

\[
S(T', D) = \sum_{k \in \partial_i} S_i \sum_{a \in \partial_{L_{k-1}} x_k^j \in [d_k]} s(S_i, x_k^j).
\]
This observation can be seen directly from Proposition 3.7. Just as was the case for DAGs in [Chickering, 2002, Lemma 7], this decomposability implies that the change in score resulting from partitioning the stages $S_i \subseteq L_{k-1}$ with associated context $X_{C_i} = x_{C_i}$ according to the outcomes of $X_j$ for some $j < k$ with $j \notin C_i$ is the same as it is for doing the same in any other CStree also containing the stage $S_i$. Hence, we can pick the tree $T$ to be any tree such that doing this partition results in the dependence model; that is, the CStree in which each node is in its own stage. Since the dependence model imposes no constraints, the result follows directly from the consistency of BIC for CStrees.

A.8. Proof of Proposition 3.9. The $p^{th}$ Bell number $B_p$ counts the number of partitions of the $p$-set $\{1, \ldots, p\}$. Equivalently, it is number of ways to divide the vertices of a $(p - 1)$-dimensional simplex into non-overlapping faces. Since CStrees are stratified, in order to enumerate them, one need only determine the number of possible ways to partition the nodes in level $k$ into stages that satisfy the condition given in Definition 3.1, take the product of these numbers with for $k = 1, \ldots, p - 1$, and then multiply by $p!$ to account for the different possible causal orderings of the variables. Hence, to get a formula for the number of CStrees on $p$ binary variables we need to determine the number of possible ways to partition the nodes in level $k$ into stages that satisfy the condition given in Definition 3.1. Since each variable is binary, there are exactly $2^k$ vertices in level $k$ of the CStree and each vertex corresponds to a unique vertex of the $k$-dimensional cube $[0,1]^k$. It follows from Definition 3.1 that the number of ways to partition the $2^k$ vertices of level $k$ of the tree so as to satisfy the conditions of a CStree corresponds to the number of ways to partition the vertices of the $k$-cube $[0,1]^k$ into non-overlapping faces; i.e., the $(k+1)^{st}$ cubical Bell number $B_{k+1}^{(c)}$.

Hence, we have proven Proposition 3.9.

A.9. Proof of Theorem 4.3. To prove the claim, we first provide a simplified version of the definition of a general interventional staged tree introduced by Duarte and Solus [2020]. Fix a nonnegative integer $j$, and let $(X_1, \ldots, X_p)$ be discrete random variables. Consider the tree $T^{(j)} = (V^{(j)}, E^{(j)})$ where $R^{(j)}_{[k]} = \{x_1^{(j)} \cdots x_k^{(j)} : x_1 \cdots x_k \in R_{[k]}\}$,

$$V^{(j)} = \{ r^{(j)} \} \cup \bigcup_{k \in [p]} R^{(j)}_{[k]},$$

$E^{(j)}$ is the union of $\{r^{(j)} \rightarrow x_1^{(j)} : x_1^{(j)} \in [d_1]\}$ and the set

$$\bigcup_{k \in [p-1]} \{x_1^{(j)} \cdots x_k^{(j)} \rightarrow x_1^{(j)} \cdots x_k^{(j)} x_{k+1}^{(j)} : x_1^{(j)} \cdots x_k^{(j)} \in R^{(j)}_{[k]}, x_{k+1}^{(j)} \in [d_{k+1}]\}.$$

For a positive integer $K$, we define a larger rooted tree $T_{[K]}$, obtained by attaching a root node $r$ to the roots of the trees $T^{(0)}, \ldots, T^{(K)}$ by an edge $r \rightarrow r^{(j)}$ for all $j \in [K]$. Given any vertex $v \in T_{[K]}$, we let $T_v = (V_v, E_v)$ denote the subtree of $T_{[K]}$ with root node $v$. The pair of subtrees $T_{r^{(i)}}, T_{r^{(j)}}$ are isomorphic (as graphs) via the isomorphism mapping $r^{(i)} \rightarrow r^{(j)}$ and $x_1^{(i)} \cdots x_k^{(i)} \rightarrow x_1^{(j)} \cdots x_k^{(j)}$ for all $k \in [p-1]$ and $x_1 \cdots x_k \in R_{[k]}$. Moreover, each of these subtrees is isomorphic to the rooted tree given by dropping all subscripts on the vertices. We let $T^c = (V^c, E^c)$ denote this tree, we refer to it as the canonical subtree of $T_{[K]}$ and to the described isomorphism as the canonical isomorphism.

Definition A.1. Let $\mathcal{I} \sqcup \mathcal{L}$ be a partitioned set of labels. A rooted tree of the form $T_{[K]} = (V, E)$, together with a labeling $\theta : E \rightarrow \mathcal{I} \sqcup \mathcal{L}$ is an interventional staged tree if

1. $\mathcal{I} = \{I_0, \ldots, I_K\}$ is a collection of subsets of $V^c \setminus \{r\}$ such that $\theta^{-1}(I_j) = \{r \rightarrow r^{(j)}\}$ for each $j \in [K]$,

2. for all $j \in [K]$ the subtree $T_{r^{(j)}} = (V_{r^{(j)}}, E_{r^{(j)}})$ is a staged tree with labeling $\theta|_{E_{r^{(j)}}} : E_{r^{(j)}} \rightarrow \mathcal{L}$, and
Figure 19. An interventional staged tree $T_I$, where $I = \{\emptyset, \{1, 2, 7, 8, 13, 14, 25, 26\}\}$. Here, we label the nodes of the canonical sub-tree $T^c$ with the letter $r$ for the root and numbers 1, ... , 30 as opposed to the outcomes of $(X_1, \ldots, X_4)$.

(3) for all pairs $i, j \in [K], \ x_1 \cdots x_{k-1} x_k \notin I_i \cap I_j$ if and only if

$$\theta(x_1^{(i)} \cdots x_{k-1}^{(i)} \rightarrow x_1^{(i)} \cdots x_k^{(i)}) = \theta(x_1^{(j)} \cdots x_{k-1}^{(j)} \rightarrow x_1^{(j)} \cdots x_k^{(j)}).$$

We will denote an interventional staged tree with labeling $\theta : E \rightarrow I \cup L$ by $T_I$ when the labels $L$ are understood from context. The set $I$ is the collection of intervention targets for the tree $T_I$. For $v \in L_1$, we will also use the notation $S_v$ for the subset that labels the edge $r \rightarrow v$ in $T_I$.

Definition A.1 is indeed a special case of the definition of an interventional staged tree presented by Duarte and Solus [2020], but it still allows for extremely granular types of intervention. See for instance, the interventional staged tree $T_I$ depicted in Figure 19. This tree has labeling inducing the staging depicted by the coloring of the nodes. The first two edges of the tree distinguish between the different distributions in the interventional setting. The top edge in the first level is labeled with the empty set, indicating that the subtree $T_r(0)$ encodes the observational distribution. The label on the bottom edge in the first level indicates the edges in the subtree $T_r(1)$ whose labels must differ from their corresponding edges in $T_r(0)$ under the canonical isomorphism. The other, non-targeted edges have their labels unchanged according to Definition A.1 (3), and this is captured by the fact that the nodes in $T_r(1)$ whose emanating edges were not targeted are in the same stage as their corresponding node in $T_1$. Note that the final column of nodes is labeled from bottom-to-top with the numbers 15, ... , 30, similar to the previous columns.

The parameter space for an interventional staged tree $T_I$ is defined to be

$$\Theta_{T_I} := \left\{ x \in \mathbb{R}^{|L|} : x_{\theta(e)} \in (0, 1) \text{ and } \sum_{e \in E(v)} x_{\theta(e)} = 1 \right\}.$$
Given an interventional staged tree $\mathcal{T}_I$ the associated *interventional staged tree model* is defined as the image of the following map

$$\psi_{\mathcal{T}_I} : \Theta_{\mathcal{T}_I} \longrightarrow \times_{k \in \{0, \ldots, K\}} \Delta^0_{\mathcal{T}_I(k)} \Delta_{\mathcal{T}_I(k)}^{K} ;$$

$$x \longmapsto \left( \prod_{e \in E(\lambda(v))} x_{\theta(e)} \right)_{v \in \mathcal{I}_T(k)}$$

Although Definition A.1 is still a special case of the definition of interventional staged tree given by Duarte and Solus [2020], it is sufficiently general so as to allow us to observe that the interventional CSTrees defined in subsection 4.2 are interventional staged trees. Indeed, interventional CSTrees have the further special property that the subtrees $\mathcal{T}_I(k)$ are not only isomorphic as rooted trees, but as staged trees. We formalize this property via the following definition.

**Definition A.2.** A graph isomorphism $\Phi : \mathcal{T} = (V, E) \longrightarrow \mathcal{T}' = (V', E')$ between two staged trees with (respective) labelings $\theta : E \longrightarrow L$ and $\theta' : E' \longrightarrow L'$ is called label-preserving if

$$\theta(v \to w) = \theta(u \to z) \iff \theta'(\Phi(v) \to \Phi(w)) = \theta'(\Phi(u) \to \Phi(z)).$$

If there exists a label-preserving isomorphism between two staged trees $\mathcal{T}$ and $\mathcal{T}'$, we say they are *isomorphic*, denoted $\mathcal{T} \cong \mathcal{T}'$.

Fix a CSTree $\mathcal{T} \in \mathcal{T}_{CS}$ and consider an interventional staged tree $\mathcal{T}_I$ with labeling $\theta$ where the interventions $I$ are such that for all $v \in L_1 \ (\mathcal{T}_v, \theta \mid E_v) \cong \mathcal{T}$. By Definition A.1, we see that the interventional staged tree $\mathcal{T}_I$ is in fact the interventional CSTree $\mathcal{T}_{(\mathcal{T}, I)}$ defined in subsection 4.2. Let $\mathcal{M}(\mathcal{T}, I)$ denote the interventional staged tree model associated to $\mathcal{T}_{(\mathcal{T}, I)}$. We then have the following proposition:

**Proposition A.1.** If $\mathcal{M}(\mathcal{T}, I)$ is the interventional staged tree model for $\mathcal{T}_{(\mathcal{T}, I)}$ then

$$\mathcal{M}(\mathcal{T}, I) = \mathcal{M}_{\mathcal{T}}(\mathcal{T}).$$

**Proof.** Let $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}(\mathcal{T}, I)$. Let $v \in L_1$ of $\mathcal{T}_I$ and set $I = S_v$. Since $\mathcal{T} \in \mathcal{T}_{CS}$ then $\mathcal{T}$ is a uniform stratified staged tree. Hence, given any edge $e$ of $\mathcal{T}$,

$$x_{\theta(e_v)} = f^{(I)}(x_k \mid x_1 \cdots x_{k-1}),$$

where $e = x_1 \cdots x_{k-1} \to x_1 \cdots x_{k-1}x_k \in E_{\mathcal{T}}$. Moreover, since $(\mathcal{T}_v, \theta \mid E_v) \cong \mathcal{T}$, there exists a label-preserving isomorphism

$$\Phi : (\mathcal{T}_v, \theta \mid E_v) \longrightarrow \mathcal{T}.$$  

Hence, since $\mathcal{T} \in \mathcal{T}_{CS}$, it follows that $f^{(I)} \in \mathcal{M}_{\mathcal{T}}$.

Suppose now that $x_1 \cdots x_{k-1}x_k \notin I \cup J$ where $I = S_v$ and $J = S_w$ for some $v, w \in L_1$ of $\mathcal{T}_{(\mathcal{T}, I)}$. Let $e_v = x_1 \cdots x_{k-1} \to x_1 \cdots x_{k-1}x_k \in \mathcal{T}_v$ and $e_w = x_1 \cdots x_{k-1} \to x_1 \cdots x_{k-1}x_k \in \mathcal{T}_w$. By Definition A.1 (2), we know that since $x_1 \cdots x_{k-1}x_k \notin I \cup J$ then $\theta(e_v) = \theta(e_w)$. So by equation (6), we know that

$$f^{(I)}(x_k \mid x_1 \cdots x_{k-1}) = f^{(J)}(x_k \mid x_1 \cdots x_{k-1})$$

whenever $x_1 \cdots x_k \notin I \cup J$. Hence, $\mathcal{M}(\mathcal{T}, I) \subseteq \mathcal{M}_{\mathcal{T}}(\mathcal{T})$.

Suppose now that $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{T}}(\mathcal{T})$. Then for each $v \in L_1$ we have that $f^{(S_v)} \in \mathcal{M}_{\mathcal{T}}$. Moreover, since $\mathcal{T}$ is uniform and stratified with $\mathcal{T} \cong (\mathcal{T}_v, \theta \mid E_v)$, we can set $x_{\theta(e_v)} = f^{(I)}(x_k \mid x_1 \cdots x_{k-1})$, where $e_v = x_1 \cdots x_{k-1} \to x_1 \cdots x_{k-1}x_k$. Then, given two $v, w \in L_1$, since

$$f^{(S_v)}(x_k \mid x_1 \cdots x_{k-1}) = f^{(S_w)}(x_k \mid x_1 \cdots x_{k-1})$$

whenever $x_1 \cdots x_{k-1} \notin S_v \cup S_w$, it follows that $x_{\theta(e_v)} = x_{\theta(e_w)}$ whenever $\theta(e_v) = \theta(e_w)$. Hence, $(f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}(\mathcal{T}, I)$, completing the proof.
As any set of valid intervention targets \( \mathcal{I} \) for a model \( \mathcal{M}_{\mathcal{I}}(\mathcal{T}) \) corresponds to a collection of stages targeted for intervention, it follows immediately that any \( \mathcal{M}_{\mathcal{I}}(\mathcal{T}) \) can be realized via the corresponding interventional staged tree model \( \mathcal{M}_{(\mathcal{T},\mathcal{I})} \). This suffices to complete the proof of Theorem 4.3.

A.10. Proof of Lemma 4.4. Let \( (f^{(I)})_{I \in \mathcal{I}} \) be a collection of distributions indexed by \( \mathcal{I} \) containing \( \emptyset \), and recall that the collection of targets \( \mathcal{I} \) associated to the interventional CStree \( \mathcal{T}_{(\mathcal{T},\mathcal{I})} \) is a collection of sets \( S_v \) indexed by the vertices \( v \in L_1 \), the first level of \( \mathcal{T}_{(\mathcal{T},\mathcal{I})} \). Suppose first that there exists \( f^{(0)} \in \mathcal{M}_\mathcal{T} \) such that for all \( v \in L_1 \) the distribution \( f^{(S_v)} \) factorizes as in (3) with respect to \( f^{(0)} \). It follows that \( f^{(S_v)} \in \mathcal{M}_\mathcal{T} \) for all \( v \in L_1 \). So it remains to show that

\[
f^{(S_v)}(x_k | x_1 \cdots x_{k-1}) = f^{(S_w)}(x_k | x_1 \cdots x_{k-1}),
\]

for any \( v, w \in L_1 \) and all \( x_1 \cdots x_{k-1} \notin S_v \cup S_w \). However, since \( f^{(S_v)} \) factorizes as in (3), we know that

\[
f^{(0)}(x_k | x_1 \cdots x_{k-1}) = f^{(S_v)}(x_k | x_1 \cdots x_{k-1})
\]

for all \( v \in L_1 \) and \( x_1 \cdots x_{k-1} \notin S_v \). Hence, for any \( v, w \in L_1 \) and \( x_1 \cdots x_{k-1} \notin S_v \cup S_w \) it must be that

\[
f^{(S_v)}(x_k | x_1 \cdots x_{k-1}) = f^{(0)}(x_k | x_1 \cdots x_{k-1}) = f^{(S_w)}(x_k | x_1 \cdots x_{k-1}).
\]

Thus, \( (f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_\mathcal{T}(\mathcal{T}) \). Conversely, suppose that \( (f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_\mathcal{T}(\mathcal{T}) \). To construct the distribution \( f^{(0)} \) with the desired properties, consider first that for each \( v \in L_1 \), the collection of nodes \( S_v \) can be partitioned according to the stages of the nodes to which they are children. Hence, we can instead think of \( S_v \) as specifying a collection of stages that are targeted under the intervention. Since we can think of stages as being targeted instead of individual nodes, we can then define the distribution \( f^{(0)} \) stage-wise; that is, we will specify the distributions \( f^{(0)}(X_k | x_1 \cdots x_{k-1}) \) for each \( x_1 \cdots x_{k-1} \in \mathcal{R}_{[k-1]} \), and each \( k \in [p] \) such that

\[
f^{(0)}(X_k | x_1 \cdots x_{k-1}) = f^{(0)}(X_k | y_1 \cdots y_{k-1})
\]

whenever \( x_1 \cdots x_{k-1} \) and \( y_1 \cdots y_{k-1} \) are in the same stage in \( \mathcal{T} \). It will then follow immediately that \( f^{(0)} \in \mathcal{M}_\mathcal{T} \).

To do this, for each \( k \in [p] \) and each stage \( S \subset L_{k-1} \), select a representative \( x_1 \cdots x_{k-1} \in S \). For this representative, note \( x_1 \cdots x_{k-1} x^1_k \notin S_v \) for a fixed \( x^1_k \in \mathcal{R}_{[k]} \) if and only if \( x_1 \cdots x_{k-1} x_k \notin S_v \) for all \( x_k \in \mathcal{R}_{[k]} \). So if \( x_1 \cdots x_{k-1} x^1_k \notin S_v \) for \( v \in L_1 \) then set \( T_S := S_v \) and

\[
f^{(0)}(X_k | y_1 \cdots y_{k-1}) := f^{(T_S)}(X_k | y_1 \cdots y_{k-1})
\]

for all \( y_1 \cdots y_{k-1} \in S \). If \( x_1 \cdots x_{k-1} x^1_k \in S_v \) for all \( v \in L_1 \) then assign \( f^{(0)}(X_k | x_1 \cdots x_{k-1}) \) arbitrarily, and set

\[
f^{(0)}(X_k | y_1 \cdots y_{k-1}) := f^{(0)}(X_k | x_1 \cdots x_{k-1})
\]

for all other \( y_1 \cdots y_{k-1} \in S \). Since these assignments have been made stage-wise, it follows that \( f^{(0)} \in \mathcal{M}_\mathcal{T} \). Moreover, for any \( x = x_1 \cdots x_p \in \mathcal{R}_{[p]} \) and any \( v \in L_1 \)

\[
f^{(S_v)}(x) = \prod_{k \in [p]} f^{(S_v)}(x_k | x_{[k-1]}),
\]

\[
= \prod_{k \in [p] \mid x_k \in S_v} f^{(S_v)}(x_k | x_{[k-1]}) \prod_{k \in [p] \mid x_k \notin S_v} f^{(T_S)}(x_k | x_{[k-1]}),
\]

\[
= \prod_{k \in [p] \mid x_k \in S_v} f^{(S_v)}(x_k | x_{[k-1]}) \prod_{k \in [p] \mid x_k \notin S_v} f^{(0)}(x_k | x_{[k-1]}).
\]

Hence, \( f^{(S_v)} \) factorizes as in (3) for some \( f^{(0)} \in \mathcal{M}_\mathcal{T} \), completing the proof.
A.11. **Proof of Proposition 4.5.** Suppose first that \((f^{(I)})_{I \in \mathcal{I}}\) satisfies the \(\mathcal{I}\)-Markov property with respect to \(\mathcal{G}^T_{\mathcal{I}}\). By Lemma 4.4, it suffices to show that each \(f^{(I)}\), for \(I \in \mathcal{I}\), factorizes as in (3). Without loss of generality, we assume that the causal ordering of \(\mathcal{T}\) is \(\pi = 12 \cdots p\). Since \((f^{(I)})_{I \in \mathcal{I}}\) satisfies the \(\mathcal{I}\)-Markov property with respect to \(\mathcal{G}^T_{\mathcal{I}}\), then we know that each \(f^{(I)}\) is Markov to \(\mathcal{G}_{\mathcal{T}}\). So by Theorem 3.3, each \(f^{(I)}\) factorizes according to \(\mathcal{T}\). Since \(\mathcal{T} \in \mathcal{T}_{CS}\), for every \(x \in \mathcal{R}_{[p]}\),

\[
 f^{(I)}(x) = \prod_{k \in [p]} f^{(I)}(x_k \mid x_{[k-1]}).
\]

Suppose that \(I = S_v\) for \(v \in L_1\), the first level of \(\mathcal{T}_{(\mathcal{T},\mathcal{I})}\), then either \(x_1 \cdots x_{k-1} x_k \in S_v\) or \(x_1 \cdots x_{k-1} x_k \notin S_v\). If \(x_1 \cdots x_{k-1} x_k \not\in S_v\), then, since \(S_v\) is complete (with respect to \(\mathcal{T}\)), we know that for every \(X_C = x_C \in \mathcal{C}_T\), either \(x_C\) is not a subcontext of \(x_1 \cdots x_{k-1}\), or \(x_C\) is a subcontext of \(x_1 \cdots x_{k-1}\) but for every \(y_1 \cdots y_{k-1}\) also having \(x_C\) as a subcontext \(y_1 \cdots y_{k-1} \not\in S_v\). Hence, for all \(X_C = x_C \in \mathcal{C}_T\), there is no \(x_1 \cdots x_{k-1} \in V_{X_C=x_C} \cap \cup_{I \in \mathcal{I}_{[k-1]}(S_v)} T\). Thus, \(w_{S_v} \to k\) is not an edge of any \(\mathcal{G}^T_{X_C=x_C}\). Hence, we know that the vertex \(w_{S_v}\) is \(d\)-separated from \(k\) given \(\text{pa}_{\mathcal{G}^T_{X_C=x_C}}(k) \cup W_{\mathcal{T}} \setminus \{w_{S_v}\}\) in \(\mathcal{G}^T_{X_C=x_C}\) for all \(X_C = x_C \in \mathcal{C}_T\). Notice now that for each \(k \in [p]\) and \(x_1 \cdots x_{k-1} \in \mathcal{R}_{[k-1]}\), there is either some \(X_C = x_C \in \mathcal{C}_T\) that is a subcontext of \(x_1 \cdots x_{k-1}\), or there is not. In the former case,

\[
 f^{(I)}(x_k \mid x_1 \cdots x_{k-1}) = f^{(I)}(x_k \mid (x_1 \cdots x_{k-1})\text{pa}_{\mathcal{G}^T_{X_C=x_C}}(k), x_C),
\]

\[
 = f^{(0)}(x_k \mid (x_1 \cdots x_{k-1})\text{pa}_{\mathcal{G}^T_{X_C=x_C}}(k), x_C),
\]

\[
 = f^{(0)}(x_k \mid x_1 \cdots x_{k-1}).
\]

Here, the first equality follows from the fact that \(f^{(I)}\) is Markov to \(\mathcal{G}_{\mathcal{T}}\). The second equality follows from Definition 4.2 (2) and the observed \(d\)-separation, and the third equality follows from the fact that if \(w_S = \emptyset\) for \(w \in L_1\) then \((T_v, \theta \mid E_v) \cong (T_w, \theta \mid E_w)\) are isomorphic staged trees. In the latter case, \(w_{S_v} \to k\) is not an edge of any interventional minimal context graph \(\mathcal{G}^T_{X_C=x_C}\) for \(X_C = x_C \in \mathcal{C}_T\), it is also not an edge of \(\mathcal{G}^T_{\emptyset}\), as \(\mathcal{I}\) is a collection of complete intervention targets. Hence, the same \(d\)-separations and invariances as above hold when taking \(X_C = x_C\) to be \(\emptyset\).

Thus, for all \(I \in \mathcal{I}\) and \(x \in \mathcal{R}_{[p]}\), we can write

\[
 f^{(I)}(x) = \prod_{k \in [p] \mid x_{[k]} \in I} f^{(I)}(x_k \mid x_1 \cdots x_{k-1}) \prod_{k \in [p] \mid x_{[k]} \notin I} f^{(0)}(x_k \mid x_1 \cdots x_{k-1}).
\]

Hence, \((f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(\mathcal{T})\).

Conversely, suppose that \((f^{(I)})_{I \in \mathcal{I}} \in \mathcal{M}_{\mathcal{I}}(\mathcal{T})\). It follows that \(f^{(I)} \in \mathcal{M}_{\mathcal{T}}\) for each \(I \in \mathcal{I}\). By Theorem 3.3, we then have that \(f^{(I)}\) is Markov to \(\mathcal{G}_{\mathcal{T}}\). Hence, for all \(X_C = x_C \in \mathcal{C}_T\), the distribution \(f^{(I)}\) entails \(X_A \perp X_B \mid X_S, X_C = x_C\) whenever \(A\) and \(B\) are \(d\)-separated given \(S\) in \(\mathcal{G}^T_{X_C=x_C}\).

To see that condition (2) of Definition 4.2 holds, fix \(I \in \mathcal{I}\) and let \(A, S \subseteq [p] \setminus C\) be disjoint subsets for which \(S \cup W_I \setminus \{w_I\}\) \(d\)-separates \(A\) and \(w_I\) in \(\mathcal{G}^T_{X_C=x_C}\). Let \(V\) denote the ancestral closure of \(A \cup S\) in \(\mathcal{G}^T_{X_C=x_C}\). Let \(B' \subseteq V\) denote the set of all nodes in \(V\) that are \(d\)-connected to \(w_I\) given \(S \cup W_I \setminus \{w_I\}\) in \(\mathcal{G}^T_{X_C=x_C}\), and set \(A' := V \setminus B' \cup S\). By applying Theorem 3.3 (3) and Lemma 4.4, the remainder of the proof follows exactly as in the proof of [Yang et al., 2018, Proposition 3.8].

A.12. **Proof of Theorem 4.6.** Suppose first that \(\mathcal{G}^T_{\mathcal{I}}\) and \(\mathcal{G}^T_{\mathcal{I}'}\) have the same skeleton and v-structures. Then, \(\mathcal{I}\) and \(\mathcal{I}'\) are compatible and there exists a bijection \(\Phi : \mathcal{I} \to \mathcal{I}'\) for which they have the same set of \(d\)-separations. So it follows from Proposition 4.5 that \(\mathcal{M}_{\mathcal{I}}(\mathcal{T}) = \mathcal{M}_{\mathcal{I}'}(\mathcal{T}')\). Hence, by Proposition A.1, it follows that \(\mathcal{T}_{(\mathcal{T},\mathcal{I})}\) and \(\mathcal{T}_{(\mathcal{T}',\mathcal{I}'')}\) are statistically equivalent.

Inversely, suppose that \(\mathcal{G}^T_{\mathcal{I}}\) and \(\mathcal{G}^T_{\mathcal{I}'}\) do not have the same skeleton and v-structures. Then either
(1) $C_T \neq C_{T'}$,
(2) $C_T = C_{T'}$, but $I$ and $I'$ are not compatible,
(3) $C_T = C_{T'}$ and $I$ and $I'$ are compatible, but there is some $X_C = x_C \in C_T$ such that $G_{X_C = x_C} \in G_T$ and $G'_{X_C = x_C} \in G_{T'}$ do not have the same skeleton and v-structures, or
(4) $C_T = C_{T'}$, $I$ and $I'$ are compatible via a bijection $\Phi : I \to I'$, but there exists $X_C = x_C \in C_T \cup \{\emptyset\}$ and a node $w_{I'}$ in $G'_{X_C = x_C}$ for which there is some $j \in [p] \setminus C$ such that $w_{I'} \to j$ is part of a v-structure in $G'_{X_C = x_C}$ but $w_{\Phi(I)} \to j$ is not part of a v-structure in $G'_{X_C = x_C}$.

For case (1), note that, by Theorem 3.5, since $C_T \neq C_{T'}$, there exists a distribution $f \in M_T$ such that $f \notin M_{T'}$. Then the interventional setting $(f^{(I)})_{I \in \mathcal{I}}$ where $f^{(I)} := f$ for all $I \in \mathcal{I}$ is an element of $M_{\mathcal{I}(T)}$ but not of $M_{\mathcal{I}(T')}$.

For case (2), suppose that $C_T = C_{T'}$, but $I$ and $I'$ are not compatible. Suppose first that $|I| \neq |I'|$. Without loss of generality, we assume $|I| < |I'|$. Then, given any $f^{(I)} \in M_{T'}$, we know $(f^{(I)})_{I \in \mathcal{I}} \in M_{\mathcal{I}(T')}$, where $f^{(I)} := f^{(0)}$ for all $I \in \mathcal{I}$. However, no sequence of distributions of this length can possibly be in $M_{\mathcal{I}(T)}$. Hence, $M_{\mathcal{I}(T)} \neq M_{\mathcal{I}(T')}$. On the other hand, suppose that $|I| = |I'|$. Since $I$ and $I'$ are not compatible, then there is no relabeling of $I'$ such that each $w_I$ has the same children in $G_{X_C = x_C}$ as it does in $G_{X_C = x_C}$ for all $X_C = x_C \in C_T$. Hence, without loss of generality, given any relabeling of $I'$ according to a bijection $\Phi : I \to I'$, there is some $X_C = x_C \in C_T$ for which there is an $I^* \in I$ and $k \in [p] \setminus C$ such that $w_{I^*} \to k$ is an edge of $G_{X_C = x_C}$ but $w_{\Phi(I^*)} \to k$ is not an edge of $G'_{X_C = x_C}$. Thus, for any relabeling of $I'$ via a bijection $\Phi : I \to I'$ there is a context $X_C = x_C \in C_T$ for which there is $I^* \in I$ and $k \in [p] \setminus C$ such that $w_{I^*}$ is d-connected to $k$ given $S := \text{pa}_{g'_{X_C = x_C}}(k)$ in $G'_{X_C = x_C}$ and d-separated from $k$ given $S$ in $G'_{X_C = x_C}$. From Duarte and Solus [2020, Section 3.1] applied to $G'_{X_C = x_C}$, the d-separation statement that holds in $G'_{X_C = x_C}$ translates into a set of polynomials $\text{Inv}_{G'_{X_C = x_C}}$ that vanish when evaluated at the points in $M_{\mathcal{I}(T')}$. Using these polynomials, we show that $M_{\mathcal{I}(T)} \neq M_{\mathcal{I}(T')}$. Suppose by way of contradiction that $M_{\mathcal{I}(T)} = M_{\mathcal{I}(T')}$.

By Duarte and Solus [2020, Theorem 4.3], $M_{\mathcal{I}(T)}$ is an irreducible variety intersected with a product of open probability simplices, $M_{\mathcal{I}(T)} = V(P(T,\mathcal{I})) \cap \Delta_{[|\mathcal{R}| - 1]}^{\otimes (k)} \times \cdots \times \Delta_{[|\mathcal{R}| - 1]}^{\otimes (k)}$, where $P(T,\mathcal{I})$ is a prime ideal in a polynomial ring and $V(P(T,\mathcal{I}))$ is the set of all points in $\mathbb{C}^{(|\mathcal{K}| + 1)|\mathcal{R}|}$ that evaluate to zero at the elements of $P(T,\mathcal{I})$. The same holds for $M_{\mathcal{I}(T')}$. Thus every polynomial in $\text{Inv}_{G'_{X_C = x_C}}$ must be satisfied by every interventional setting in $M_{\mathcal{I}(T')}$.

For case (3), suppose that $C_T = C_{T'}$ and $I$ and $I'$ are compatible. Hence, we may relabel $I'$ so that all nodes in $G'_{T'}$ and $G_T$ have the same labels and, after this relabeling, all nodes $w_I$ have the same children in $G'_{X_C = x_C}$ and $G_{X_C = x_C}$ for all $X_C = x_C$. Suppose now that there exists $X_C = x_C \in C_T$ such that $G_{X_C = x_C} \in G_T$ and $G'_{X_C = x_C} \in G_{T'}$ do not have the same skeleton and v-structures. In this case, it follows from Theorem 3.5 that $T$ and $T'$ are not statistically equivalent. That is, $M_T \neq M_{T'}$. So, without loss of generality, there exists $f \in M_T$ such that $f \notin M_{T'}$. By setting $f^{(I)} := f$ for all $I \in \mathcal{I}$, we produce $(f^{(I)})_{I \in \mathcal{I}}$ that is in $M_{\mathcal{I}(T)}$ but not in $M_{\mathcal{I}(T')}$. Hence, $M_{\mathcal{I}(T)} \neq M_{\mathcal{I}(T')}$. Finally, in case (4), we assume that $C_T = C_{T'}$ and $I$ and $I'$ are compatible. As in case (3), these assumptions ensure that we may relabel $I'$ so that all nodes in $G'_{T'}$ and $G_T$ have the same labels and, after this relabeling, all nodes $w_I$ have the same children in $G'_{X_C = x_C}$ and $G_{X_C = x_C}$ for
all \( X_C = x_C \). It follows that, after this relabeling, \( G_{X_C = x_C}^I = \mathcal{G}_{X_C = x_C}^I \) and \( G_{X_C = x_C}^{I'} = \mathcal{G}_{X_C = x_C}^{I'} \) have the same skeleton for all \( X_C = x_C \in \mathcal{C}_T \). Hence, if we assume now that there is some \( X_C = x_C \in \mathcal{C}_T \) for which there is a node \( w_I^* \) in \( G_{X_C = x_C}^I \) and \( k \in [p] \setminus C \) for which \( w_I^* \rightarrow j \) is part of a v-structure in \( G_{X_C = x_C}^I \) but \( w_I^* \rightarrow j \) is not part of a v-structure in \( G_{X_C = x_C}^{I'} \), then it must be that there exists \( k \in [p] \setminus (C \cup \{j\}) \) such that \( w_I^* \rightarrow j \leftarrow k \) is a v-structure in \( G_{X_C = x_C}^I \) but \( w_I^* \rightarrow j \rightarrow k \) in \( G_{X_C = x_C}^{I'} \). Given such a scenario, let \( S := \text{pa}_{G_{X_C = x_C}^I}(k) \). It follows that \( w_I^* \) is d-separated from \( k \) given \( S \) in \( G_{X_C = x_C}^I \) but d-connected given \( S \) in \( G_{X_C = x_C}^{I'} \). Similar to the argument given in case (2), the d-separation statement that holds in \( G_{X_C = x_C}^I \) translates into a set of polynomials \( \text{Inv}_{G_{X_C = x_C}^I} \) that vanish when evaluated at the points in \( \mathcal{M}_I(T) \). Supposing for the sake of contradiction that \( \mathcal{M}_I(T) = \mathcal{M}_I'(T') \), the same argument shows that the polynomials in \( \text{Inv}_{G_{X_C = x_C}^{I'}} \) must also vanish on all points in the model \( \mathcal{M}_I'(T') \), which would contradict \( w_I^* \) and \( k \) being d-connected given \( S \) in \( G_{X_C = x_C}^{I'} \). Hence, we conclude that \( \mathcal{M}_I(T) \neq \mathcal{M}_I'(T') \), which completes the proof.

**Appendix B. Additional Material for the Real Data Analysis**

This appendix contains additional figures supplementing Section 6.2. Namely, Figure 20 shows the three CStrees that are statistically equivalent to the BIC-optimal CStree depicted in Figure 16, and Figures 21 and 22 show the four BIC-optimal interventional CStrees whose contexts graphs are depicted in Figure 18.

**Figure 20.** The three CStrees that are statistically equivalent to the tree depicted in Figure 16.
Figure 21. Two of the four BIC-optimal interventional CStrees learned in Section 6.2.2.
Figure 22. Two of the four BIC-optimal interventional CStrees learned in Section 6.2.2.
REFERENCES

D. Andersson, S. Madigan and M. Perlman. A characterization of Markov equivalence classes for acyclic digraphs. *Ann. Statist.*, 25(2):505–541, 1997. ISSN 0090-5364. doi: 10.1214/aos/1031833662. URL https://doi.org/10.1214/aos/1031833662. 5, 27

C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller. Context-specific independence in Bayesian networks. In *Uncertainty in artificial intelligence (Portland, OR, 1996)*, pages 115–123. Morgan Kaufmann, San Francisco, CA, 1996. 1, 5

F. Carli, M. Leonelli, E. Riccomagno, and G. Varando. The r package stagedtrees for structural learning of stratified staged trees. *arXiv preprint arXiv:2004.06459*, 2020. 20

D. M. Chickering. A transformational characterization of equivalent bayesian network structures. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI’95, page 87–98, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603859. 5, 27

D. M. Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3:507–554, 2002. 11, 27, 31, 32

D. M. Chickering, D. Heckerman, and C. Meek. A bayesian approach to learning bayesian networks with local structure. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, UAI’97, page 80–89, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc. ISBN 1558604855. 5

R. Collazo, C. Görgen, and J. Q. Smith. *Chain event graphs*. Chapman & Hall/CRC Computer Science and Data Analysis Series. CRC Press, Boca Raton, FL, 2018. ISBN 978-1-4987-2960-4. 4

R. Cowell and J. Smith. Causal discovery through MAP selection of stratified chain event graphs. *Electron. J. Stat.*, 8(1):965–997, 2014. doi: 10.1214/14-EJS917. URL https://doi.org/10.1214/14-EJS917. 12

D. Dua and C. Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml. 22, 23

E. Duarte and C. Görgen. Equations defining probability tree models. *J. Symbolic Comput.*, 99:127–146, 2020. ISSN 0747-7171. doi: 10.1016/j.jsc.2019.04.001. URL https://doi.org/10.1016/j.jsc.2019.04.001. 8, 30

E. Duarte and L. Solus. Algebraic geometry of discrete interventional models. *Preprint available at https://arxiv.org/abs/2012.03593*, 2020. 3, 4, 5, 12, 14, 31, 32, 33, 34, 37

E. Duarte and L. Solus. A new characterization of discrete decomposable models. *Preprint available at https://arxiv.org/abs/2105.05907*, 2021a. 6, 8, 14

E. Duarte and L. Solus. Cstrees. https://github.com/soluslab, 2021b. 20, 22

E. Duarte, O. Marigliano, and B. Sturmfels. Discrete statistical models with rational maximum likelihood estimator. *Bernoulli*, 27(1):135–154, 2021. ISSN 1350-7265. doi: 10.3150/20-BEJ1231. URL https://doi.org/10.3150/20-BEJ1231. 10, 31

F. Eberhardt, C. Glymour, and R. Scheines. On the number of experiments sufficient and in the worst case necessary to identify all causal relations among n variables. In *In the Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, pages 178–184, 2005. 13

N. Friedman and M. Goldszmidt. Learning bayesian networks with local structure. In *In Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence*, pages 252–262, 1996. 5

D. Geiger and D. Heckerman. Knowledge representation and inference in similarity networks and Bayesian multinets. *Artificial Intelligence*, 82(1-2):45–74, 1996. ISSN 0004-3702. doi: 10.1016/0004-3702(95)00014-3. URL https://doi.org/10.1016/0004-3702(95)00014-3. 1

D. Geiger, D. Heckerman, H. King, and C. Meek. Stratified exponential families: graphical models and model selection. *Ann. Statist.*, 29(2):505–529, 2001. ISSN 0090-5364. doi: 10.1214/aos/1009210550. URL https://doi.org/10.1214/aos/1009210550. 11
L. Solus, Y. Wang, and C. Uhler. Consistency Guarantees for Greedy Permutation-Based Causal Inference Algorithms. *Biometrika*, 01 2021. ISSN 0006-3444. doi: 10.1093/biomet/asaa104. URL https://doi.org/10.1093/biomet/asaa104. 27

P. Spirtes, C. Glymour, and R. Scheines. *Causation, prediction, and search*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, second edition, 2000. ISBN 0-262-19440-6. With additional material by David Heckerman, Christopher Meek, Gregory F. Cooper and Thomas Richardson, A Bradford Book. 27

Seth Sullivant. *Algebraic statistics*, volume 194. American Mathematical Soc., 2018. 30

P. Thwaites. Chain event graphs: Theory and applications. *Dissertation University of Warwick*, 2008. 12

Peter Thwaites, Jim Q. Smith, and Eva Riccomagno. Causal analysis with chain event graphs. *Artificial Intelligence*, 174(12-13):889–909, 2010. ISSN 0004-3702. doi: 10.1016/j.artint.2010.05.004. URL https://doi.org/10.1016/j.artint.2010.05.004. 12

T. Verma and J. Pearl. Causal networks: semantics and expressiveness. In *Uncertainty in artificial intelligence, 4*, volume 9 of *Mach. Intelligence Pattern Recogn.*, pages 69–76. North-Holland, Amsterdam, 1990a. doi: 10.1016/B978-0-444-88650-7.50011-1. URL https://doi.org/10.1016/B978-0-444-88650-7.50011-1. 5

Thomas Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*, UAI ’90, page 255–270, USA, 1990b. Elsevier Science Inc. ISBN 0444892648. 10, 18, 27

Y. Wang, L. Solus, K. D. Yang, and C. Uhler. Permutation-based causal inference algorithms with interventions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS’17, page 5824–5833, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. 3, 12, 27

K. D Yang, A. Katcoff, and C. Uhler. Characterizing and learning equivalence classes of causal dags under interventions. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5541–5550. PMLR, 10–15 Jul 2018. URL http://proceedings.mlr.press/v80/yang18a.html. 3, 12, 13, 15, 17, 18, 19, 27, 36

Departamento de Matemática, Faculdade de Ciências da Universidade do Porto, Rua do Campo Alegre, 4169-007 Porto, Portugal

Email address: eliana.gelvez@fc.up.pt

Institutionen för Matematik, KTH, SE-100 44 Stockholm, Sweden

Email address: solus@kth.se