THE CAUSAL MANIPULATION AND BAYESIAN ESTIMATION OF CHAIN EVENT GRAPHS

By Eva Riccomagno

Department of Mathematics, Polytechnic of Turin

AND

By Jim Q. Smith

Department of Statistics, The University of Warwick

Discrete Bayesian Networks (BNs) have been very successful as a framework both for inference and for expressing certain causal hypotheses. In this paper we present a class of graphical models called the chain event graph (CEG) models, that generalises the class of discrete BN models. This class is suited for representing conditional independence and sample space structures of asymmetric models. It retains many useful properties of discrete BNs, in particular admitting conjugate estimation. It provides a flexible and expressive framework for representing and analysing the implications of causal hypotheses, expressed in terms of the effects of a manipulation of the generating underlying system. We prove that, as for a BN, identifiability analyses of causal effects can be performed through examining the topology of the CEG graph, leading to theorems analogous to the Backdoor theorem for the BN.

1. Introduction

Bayesian networks have now been extended to Causal Bayesian Networks (CBNs) using a non-parametric representation based on structural equation models [14, 26, 27, 43]. These provide a framework for expressing assertions about what might happen when the system under study is externally manipulated and some of its variables are assigned certain values. Recently the desirability of CBN models have been vigorously debated for example against the classes of models based on counterfactuals [32, 35] and [10, 28, 33, 36]. It seems to us that the extension of a BN to a CBN depends strongly on three assumptions:

1. the existence of information about a background *idle* (unmanipulated) system,
2. the existence of a network of simulators, or equivalently a collection of data generating processes, that can be used to generate conditional independences [27, 28, 43] and that is believed to appropriately model the process of interest,
3. the belief that manipulation of one or more simulators can model an external intervention on the underlying process being modelled.

Keywords and phrases: Backdoor theorem, Bayesian networks, Event trees, Causal graphical models, Probability estimation, Chain event graph.

The appropriateness of the adopted definition of a cause as system manipulation is highly contentious in general and will only be touched upon in the conclusion of the paper (see [16, 28]). But, as a description of a manipulation of a simulator network it appears very natural. Thus this paper assumes that the simulator analogy in Items 2. and 3. above is valid. In particular the term "causal" will apply to the modelling simulator network and *not* to the underlying process being modelled.

Even within the scope of the simulator analogy it is not clear that a definition of causality in terms of a BN is as appropriate or as general as it might be. Motivated by comments in [36], we develop an alternative graphical representation of causal models, called *chain event graph model*. This is constructed from an event tree together with a set of exchangeability assumptions. It can be seen as a generalisation of a probability graph [4, 36] and typically has many less nodes than the original event tree. It was introduced in [40] in parallel with the present paper. Within the class of CEG models, for any given application, an analogue of the causal extension used to transform a BN into a CBN is transparent and is as compelling as it is for a BN. This class thus relaxes the hypothesis in Item 1. above, whilst retaining the hypotheses in Items 2. and 3.

There are several technical reasons why chain event graph models are important. Throughout the paper we motivate our results with comparison and analogies to BNs and CBNs.

The first one is entirely practical. Many observed systems can be elegantly modelled by a BN, but many other processes arising from, for example genomic, epidemiology, multi-agent systems, cannot be so conveniently and fully described by a BN. For examples of these see [3, 22, 29]. Despite the proliferation of graphical models over the last two decades, the first stage of the elicitation of a model can still be based on the elicitation of an event tree. This happens for example in Bayesian decision analysis [13], risk analysis [2], physics [20], biological regulation [7]. Although often topologically vast, event trees have several advantages over BNs including: *(i.)* they explicitly acknowledge asymmetries embedded in a structure both in its development and in its sample space structure, *(ii.)* their semantics are much closer to many verbal descriptions of the world: especially when that description revolves round how things happen rather than how the world appears. These advantages are compellingly argued in e.g. [36] and [27, 43] in the context of "causality". Various methods for interrogating an elicited model and for forming a framework for propagation have been developed in the recent years. In particular [12, 18] discuss probability decision graphs to embody sets of conditional independence statements and to give an explicit representation of the sample space of the problem. Essentially they depict state transitions in the study of discrete stochastic processes. Context-specific networks [3] supplement the BN with additional structure often via trees, [29] use confactors to study propagation even outside tree structures, [24] develop methods based on case-factor diagram. For a comparison of these representations with CEGs we refer to the

introduction in [40].

A second reason is that by using the framework of event trees, the definition of manipulative cause is freed from the shackles of the conditional independence relations imposed by the restrictive class of BN models and allows direct analogues to BNs of, for example, total causal effect analysis [26]. Whilst the graph of a BN expresses conditional independence statements explicitly, event trees and probability trees depict relationships between state spaces directly. In probability trees conditional independence relations can be embedded through equations linking probabilities labelling the edges of the tree. These are used to construct the vertices of the CEG and its undirected edges. Analogues to d-separation theorems that give sufficient conditions for determining whether a conditional independence statement holds, are given for CEG in [40]. Recently separation theorems associated with other graphical models have also been derived [9, 19, 25, 27].

Third, the flexibility of a framework based on event trees separates the causal hypotheses (Item 3.) from any direct link with the measurement process (Item 2.). One problem with some graphical models is that they take a collection of measurement random variables as a given and express and manipulate conditional independence properties round these. But if a model embodies hypotheses about how situations might unfold, it is often not obvious how to define random variables whose mutual relationships might express this unfolding. CEG models can be used to automatically construct random vectors whose conditional independence structure represents the relationships implied by an event tree description of a process.

Fourth, it has recently been noted that the dimension of the sample space is critical for determining identifiability especially when many variables are hidden, even in symmetric models [37, 38, 41]. This cannot be expressed explicitly in a BN but it is expressed in an event tree and is retained in the CEG. Notice that the sample space given by a tree is not necessarily of product form as in a BN.

The next examples show that together with those technical reasons there are other compelling modelling reasons to develop CEG models.

Nowadays networks of simulators exist that purport to model environmental catastrophes. One such network was supported under the RODOS project [39]. Usually a simulator in the RODOS network is extremely complex, often with a randomising component, so that simulation samples are costly. Various scenarios can be played out through the system but accurate margins of the variables are difficult to access reliably. Each simulator is owned by an agent who will give only the output and not the internal algorithms so that the only data available are the values obtained from each simulator given certain configurations of its inputs. In fact, this and prior experimental information about individual mechanisms modelled by the simulators is all that is available for inference. Data on how the process proceeds through the real network is rarely available: most types of accident have not yet taken place (thankfully). So understanding the workings of the simulator network is often

as close as we can get in an empirical study.

Different scenarios will lead to different simulators of the network to be active. For example when modelling a nuclear accident with an associated release of contaminating substances in the atmosphere, if there is no release then the only simulators that are enacted are the possible precautionary countermeasures that might be put in place and the module evaluating the economic cost of these. If there is a release and the countermeasure of a carefully enforced total food ban is put in place, then the human intake module of the network will not depend on the absorption of contamination in food, and so on. So what happens earlier determines not only the value but the nature of what happens subsequently. A BN gives a useful and compact summary of some of the network information, but it cannot express graphically all the context specific network information whereas a probability tree can. In this respect it is safer and more comprehensive to build definitions of causal effects around the event tree rather than the BN.

Not all simulators are manipulable in the sense that any manipulation can be given a real interpretation. Furthermore the types of manipulations that are of interest are likely to be incremental modifications to certain countermeasures in certain contingencies, like including an additional area in an evacuation policy, not wholesale uniform change as addressed in a causal BN.

Another class of examples where the tree is often a better representation of an underlying process is in models of biological regulatory mechanisms. Often such mechanisms are highly asymmetrical and context specific, typically containing many noisy "and" and "or" gates. The simulator analogy seems to work well in this class of examples [1]. In particular asymmetric manipulations, such as attaching a virus to a gene to enforce overexpression, would appear to preserve this analogy when it is extended to the manipulated system.

Although effects of a cause can be reasonably represented by a random variable, at times the specification of a cause as the value of a random variable can be artificial. For example, suppose interest focuses on the effect of inspection frequency on numbers of derailments of goods trains: our cause. Why is it necessary to construct a random variable representing *all* possible inspection frequencies (with an associated distribution) when interest lies only in the relative merits of a rail track inspection program with 3 monthly checks as against 6 monthly checks (or even just one of these)? It seems perverse to define causality in a way that demands this unnecessary level of specification. At the root of the problem here and in the examples cited above is the fact that causes are more naturally represented as conditioning *events* than as random variables. Such conditioning is not elegantly expressed in the BN but is simply and intrinsically described in a probability tree and the derived CEG. Analogous arguments are made by Dawid [10] who argues that causes are decisions and not decision rules.

Finally, in an event tree the postulated causal mechanism is made more explicit by relating the predictions to hypotheses about how the observer believes things might happen. This is appealing from the Bayesian perspective

because it admits the formal inclusion of the observer into the inference, contrasting with the implicit and arguably misleading apparent determinism of the CBN which is based on spuriously objective collections of measurements.

For a good discussion of many of the above points see [36], in particular on the advantages of event trees for coding asymmetrical problems, as powerful expression of an observer's beliefs especially when those beliefs are based on an underlying conjecture about a causal mechanism

Section 2 contains the basic terminology and definitions. In Section 3 we show that as discrete faithful BNs (see [9, 17, 42] for observed systems and [8] for manipulated systems) CEG models admit a conjugate multinomial-Dirichlet prior to posterior analysis when hypotheses are about populations of exchangeable units. The extension from event trees and CEGs to causal probability trees and causal CEG is made in Section 4. A theorem of identifiability analogous to the Backdoor theorem is proved in Section 5.

2. Simulating with Probability Trees

From a Bayesian perspective probability trees describe the observer's beliefs about what will happen as events unfold. From standard probability theory the edges out of a node v of the probability tree represent the possible unfolding that can occur from the situation labelled by v, or equivalently the event space of a random variable that can be indexed by v. The sample space of the experiment at a node is given by the branches of the probability tree at that point step.

Through two equivalence relations on the vertices of the event tree, we construct a new model structure, called a chain event graph, which includes Bayesian Networks and which provides a natural framework for defining causality.

We start with some definitions to set up notation and formalise ideas. Some of these definitions are slightly non-standard for reasons that will become apparent later in the paper. Section 2.1 draws strongly from [36, 40] and we refer to those works for further details.

2.1. Probability trees

2.1.1. Graphs

Definition 1 *A* (rooted mixed) graph $\mathcal{G} = (V(\mathcal{G}), E_d(\mathcal{G}), E_u(\mathcal{G}))$ *consists of*

1. $V(\mathcal{G}) = \{v_0, v_1, \ldots, v_n\}$ *a finite set of* vertices *or* nodes*,*
2. $E_u(\mathcal{G}) = \{\{v, v'\} : v, v' \in V(\mathcal{G})$ *and* $v \neq v'\}$ *a finite set of* undirected edges *and*
3. $E_d(\mathcal{G}) = \{e = (v, v') : v, v' \in V(\mathcal{G})$ *and* $v \neq v'\}$ *a multiset of ordered pairs of vertices and two maps* $\mathrm{pa}, \mathrm{ch} : E_d(\mathcal{G}) \to V(\mathcal{G})$ *where* $\mathrm{pa}(v, v') = \mathrm{pa}(e) = v$ *is the* parent *of* v' *and* $\mathrm{ch}(v, v') = \mathrm{ch}(e) = v'$ *is a* child *of* v*.* $E_d(\mathcal{G})$ *is the multiset of* directed edges *of* $\mathcal{G}$*.*

Sometimes we use the notation (V, E_d, E_u) for $\mathcal{G}$. Note the following.

- We assume that exactly one vertex $v_0 \in V(\mathcal{G})$ has no parent and call it the *root vertex*.
- A graph where E_u is the empty set is called a *directed graph*, sometimes written $\mathcal{G} = (V(\mathcal{G}), E(\mathcal{G}), \mathrm{pa}, \mathrm{ch})$.
- A *(directed) tree* is a directed graph in which all vertices except the root vertex have exactly one parent. A vertex of a tree with no child is called a *leaf*. We write $\mathcal{T} = (V(\mathcal{T}), E(\mathcal{T})) = (V, E)$.
- A *(purely directed) path* between two vertices v_i and v_j in a graph is a sequence of directed edges $\lambda = \lambda(v_i, v_j) = (e_1, \ldots, e_{n[\lambda]})$ where $\mathrm{pa}(e_1) = v_i$, $\mathrm{ch}(e_{n[\lambda]}) = v_j$, $\mathrm{ch}(e_k) = \mathrm{pa}(e_{k+1})$ for $k = 1, \ldots, n[\lambda] - 1$. The number of edges in the path is called the *length* of the path and it is $n[\lambda]$.
- We also write $v \in \lambda$ when the path λ passes through the vertex v.
- A graph $\mathcal{G} = (V, E_d, E_u)$ is said to have *coloured directed edges* if there is a non-trivial partition $\{E_d^{(i)}(\mathcal{G}) : 1 \leq i \leq K\}$ of E_d with K integer, $K \geq 1$. If $e, e' \in E_d^{(i)}(\mathcal{G})$ then e and e' are said to *have the same colour* i, $1 \leq i \leq K$.

Definition 2 *Two graphs $\mathcal{G}_1 = (V(\mathcal{G}_1), E_d(\mathcal{G}_1), E_u(\mathcal{G}_1))$ with associated maps $\mathrm{pa}_1, \mathrm{ch}_1$ and $\mathcal{G}_2 = (V(\mathcal{G}_2), E_d(\mathcal{G}_2), E_u(\mathcal{G}_2))$ with $\mathrm{pa}_2, \mathrm{ch}_2$ are isomorphic if there exists a one-to-one map $\mu : V(\mathcal{G}_1) \to V(\mathcal{G}_2)$ such that*

1. *$(\mu(v), \mu(v')) \in E_d(\mathcal{G}_2)$ if and only if $(v, v') \in E_d(\mathcal{G}_1)$,*
2. *$\mathrm{pa}_2(\mu(v), \mu(v')) = \mu(v)$ if and only if $\mathrm{pa}_1(v, v') = v$,*
3. *$\mathrm{ch}_2(\mu(v), \mu(v')) = \mu(v')$ if and only if $\mathrm{ch}_1(v, v') = v'$ and*
4. *$\{\mu(v), \mu(v')\} \in E_u(\mathcal{G}_2)$ if and only if $\{v, v'\} \in E_u(\mathcal{G}_1)$.*

In this paper only rooted mixed graphs and purely directed paths are of interest and so the qualifiers "rooted mixed" and "purely directed" will henceforth be omitted.

2.1.2. Trees

Let $\mathcal{T} = (V(\mathcal{T}), E(\mathcal{T}))$ be a directed tree. The set $S(\mathcal{T})$ of non-leaf vertices is called the set of *situations* of $\mathcal{T}$ and has particular significance.

Definition 3 *Let $\mathbb{X} = \{\lambda(v_0, v) : v \in V(\mathcal{T}) \backslash S(\mathcal{T})\}$ be the set of root-to-leaf paths. Elements of $\mathbb{X}$ are called* atomic events. *For $v \in S(\mathcal{T})$ let $\mathbb{X}(v)$ be the set of children of v.*

Note that

- $\mathbb{X}(v)$ can be seen equivalently as the set of edges out of v and that $\{\mathbb{X}(v) : v \in S(\mathcal{T})\} \cup \{v_0\}$ partition $V(\mathcal{T})$.
- $\mathbb{X}$ is in one-to-one correspondence with the leaves of the tree.
- A partial order on $V(\mathcal{T})$ is determined by the paths. Heuristically, if there is a path from v to v' then v' cannot happen before v and v' follows v in the partial order induced by the paths.

The situations along each root-to-leaf path correspond to a possible historical development of the problem we are modelling. The directionality in

the tree is natural and expresses a conjecture about the ordering in which one situation follows another (see in particular [36, Section 2.8]). Later this directionality will play the role of a causal (partial) ordering.

A *probability tree* is a directed tree such that to each situation $v \in S(\mathcal{T})$ is associated a discrete random variable $X(v)$ whose sample space is $\mathbb{X}(v)$. The basic measurable space is the one given by the set of atomic events endowed with the path σ-algebra. The random variable $X(v)$ can be defined conditional on having reached the position v. Thus for $\lambda \in \mathbb{X}$ $X(v)(\lambda) = v'$ is equivalent to say that $X(v)$ maps λ into the set of paths through v and v'. Existence and uniqueness of a joint probability distribution for the full tree follows from standard probability theory.

The distribution of $X(v)$, $v \in S(\mathcal{T})$, is determined by the *primitive probabilities* $\pi(v'|v) = P(X(v) = v')$ for $v' \in \mathbb{X}(v)$.

Moreover the random variables on vertices along a path are required to be mutually independent.

The primitive probability $\pi(v'|v)$ is a colour for the directed edge $e = (v, v')$, thus we shall write $\pi(e) = \pi(v'|v)$ as well. If $\pi(v'|v) = 0$ for some v, v' then the branch starting at v' can be deleted from the tree as any atomic event including v' has zero probability of occurring.

Example 4 To explain notation, Figure 1 shows the probability tree for two binary random variables X and Y with X happening before Y and joint law $P(X = x, Y = y)$ for $x, y \in \{0, 1\}$. We have $X(v_0) = X$, $X(v_1) = [Y|X = 0]$, $X(v_2) = [Y|X = 1]$ and $\pi(v_1|v_0) = P(X = 0)$, $\pi(v_3|v_1) = P(Y = 0|X = 0)$. The recursive formula for a joint probability corresponds to the independence of random variables along a path, e.g. $P(X = 0, Y = 0) = \pi(v_1|v_0)\pi(v_3|v_1)$ as $X(v_0)$ and $X(v_1)$ are independent. No statement is made about the dependence relation between random variables on nodes on different paths.

The interpretation of the random variables $\{X(v), v \in S(\mathcal{T})\}$ is clearest when the probability tree describes the paths taken through a network of simulators. A sample from the network begins with the root situation v_0 that is an *active* random variable $X(v_0)$. The simulator produces an output v_0' equivalently $X(v_0) = v_0' \in \mathbb{X}(v)$ with some probability; the simulator located at v_0' is activated and a value from the random variable $X(v_0')$ is drawn. The process is repeated until a leaf node is reached. Only one simulator is active at each step. Thus a draw from the whole simulator network corresponds to a root-to-leaf path of the probability tree: a point we utilize in Section 4. In particular, each sample unit runs across a single path of the tree.

Definition 5 *Two situations $v_1, v_2 \in \mathcal{S}(\mathcal{T})$ are* stage-equivalent *if and only if*

1. *there exists a one-to-one map $\mu : \mathbb{X}(v_1) \to \mathbb{X}(v_2)$ and*
2. *the distributions of $X(v_1)$ and $X(v_2)$ are the same consistently with μ, that is $\pi(v|v_1) = \pi(\mu(v)|v_2)$ for all $v \in \mathbb{X}(v_1)$.*

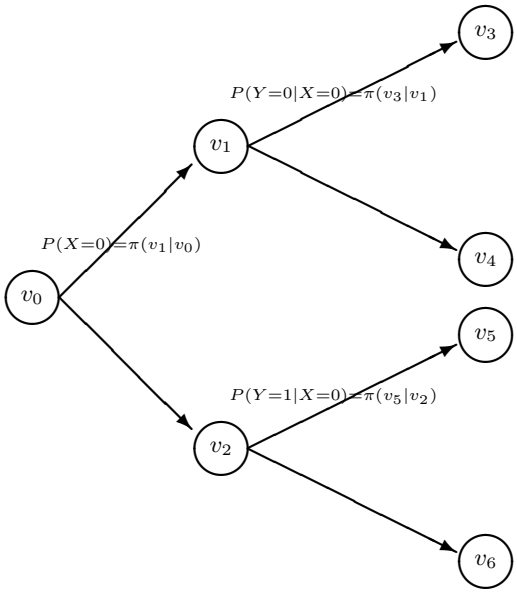

FIG 1. *Probability tree for Example 4*

The corresponding equivalence classes are called stages *and* $L(\mathcal{T})$ *is the set of stages.*

For each stage $u \in L(\mathcal{T})$ *define*

$$\Pi(u) = \{\pi(v'|v) : v' \in \mathbb{X}(v) \text{ for some } v \text{ representative of } u\}$$

and $\Pi(\mathcal{T}) = \bigcup_{u \in L(\mathcal{T})} \Pi(u)$.

The pair $(\mathcal{T}, \Pi(\mathcal{T}))$ *is called a* probability tree model.

The set of primitive probabilities $\Pi(\mathcal{T})$ is no larger than the set of all primitive probabilities and clearly still sufficient for a complete description of all distributions defined on the probability tree.

The probability $\pi(\lambda)$ of an atomic event $\lambda \in \mathbb{X}$ can now be given as products of the primitive probabilities in $\Pi(\mathcal{T})$. Let $\lambda = (e_1, \ldots, e_{n[\lambda]}) \in \mathbb{X}$ be the path from the root v_0 to the leaf vertex $\mathrm{ch}(e_{n[\lambda]})$, where $n[\lambda] \geq 1$ is assumed. Then

$$\pi(\lambda) = \prod_{j=1}^{n[\lambda]} \pi(e_j) = \prod_{j=1}^{n[\lambda]} \pi(\mathrm{ch}(e_j)|\,\mathrm{pa}(e_j)) \tag{1}$$

where $\pi(e_j) \in \Pi(\mathrm{pa}(e_j))$. The sum-to-one condition gives $\sum_{\lambda \in \mathbb{X}} \pi(\lambda) = 1$ together with $\sum_{v' \in \mathbb{X}(v)} \pi(v'|v) = 1$ for all $v \in \mathcal{S}(\mathcal{T})$.

Stages express conditional independence statements by stating that if two situations are in the same stage, then their probability distributions are the same.

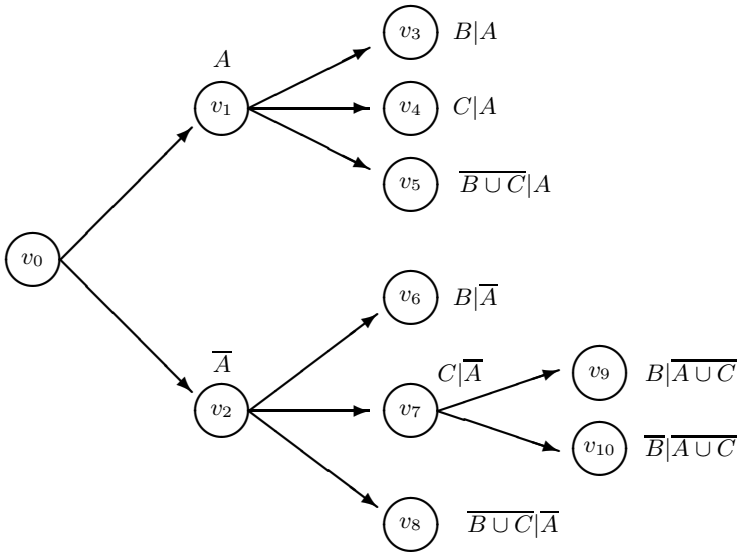

FIG 2. *Stages and independence.*

Example 6 (cont. Example 4) In Figure 1 if v_1 and v_2 are in the same stage then two cases can occur according to whether v_3 maps onto v_5 or onto v_6. Either $P(Y = 1|X = 0) = P(Y = 1|X = 1)$ i.e. X and Y are independent or $P(Y = 0|X = 0) = P(Y = 1|X = 1)$ i.e. $P(Y = 0|X = 0) = P(X = Y = 0) + P(X = Y = 1)$. If the values X and Y take were -1 and 1 then this last equality would imply the independence between X and XY [38].

For an analysis of the relationship between stages and independence see [40]. Here we just consider a small example. Only three events A, B, C can occur and their joint history unfolds according to the probability tree in Figure 2. The independence between A and B corresponds to $\pi(v_3|v_1)\pi(v_1|v_0) = [\pi(v_6|v_2) + \pi(v_9|v_7)\pi(v_7|v_2)]\,\pi(v_2|v_0)$. If v_7 were a leaf node then v_1 and v_2 would be in the same stage with $\pi(v_3|v_1) = \pi(v_6|v_2)$ and $\pi(v_4|v_1) = \pi(v_7|v_2)$ if and only if A and B would be independent events.

2.2. Chain event graphs

Probability trees depict the structure of a state space but cannot express graphically, as a BN can, any relationships between the underlying random variables. In this section we assume that the observer is able to express two pieces of qualitative information: the topology of the probability tree and its stages. We will show that these two sources of information can be fully represented using a mixed graph with coloured edges called a chain event graph. The colouring of this graph can then be used to read off conditional independence statements associated with various random variables measurable with respect to the path σ-algebra of the underlying probability tree model.

Suppose that two units have arrived at different situations v and v^* of a tree and that the processes governing their subsequent evolutions is believed to be identical. For this to happen it is necessary that the sub-trees $\mathcal{T}(v)$ beginning at v and $\mathcal{T}(v^*)$ with root v^* are topologically isomorphic, in particular their vertices and edges can be identified. Moreover the non-leaf vertices of these two sub-trees are in the same stage. In [19] $\mathcal{T}(v)$ is called the subgraph induced by v and its ancestors.

Definition 7 *Let $\mathcal{T} = (V(\mathcal{T}), E(\mathcal{T}))$ be a probability tree model. For $v, v^* \in S(\mathcal{T})$ define $\mathcal{T}(v)$ ($\mathcal{T}(v^*)$) to be the sub-tree of $\mathcal{T}$ starting at v (v^*) respectively. We say that v and v^* are equivalent if and only if*

1. *$\mathcal{T}(v)$ and $\mathcal{T}(v^*)$ are isomorphic. Let μ be the map in Definition 2.*
2. *For every w non-leaf vertex in $\mathcal{T}(v)$, w and $\mu(w)$ are in the same stage and*
3. *$\pi(v_2|v_1) = \pi(\mu(v_2)|\mu(v_1))$ for all possible $v_1, v_2 \in \mathcal{T}(v)$.*

The induced equivalence classes are called positions *and $K(\mathcal{T})$ is the set of positions.*

Note that this is a predictive, not a retrospective equivalence. Once a unit reaches the vertex v or the vertex v^* all pairs of possible unfoldings from v and v^* have the same probabilities. Thus two situations have the same position when their future evolutions are governed by collections of random variables with the same distribution. It is in this respect that positions are natural objects on which to describe a causal manipulation.

Clearly the partition of situations into position is a coarsening of that into stages.

Positions are used to form the vertices of a new graph called the chain event graph. This is a mixed graph whose undirected edges join positions at the same stage and whose directed paths correspond to root-to-leaf paths of the probability tree model.

Definition 8 *Let $(\mathcal{T}, \Pi(\mathcal{T}))$ be a probability tree model. Its chain event graph, $\mathcal{C}(\mathcal{T})$, is the mixed graph with coloured edges defined as follows.*

1. *The vertex set is $V(\mathcal{C}(\mathcal{T})) = K(\mathcal{T}) \cup \{w_\infty\}$. The vertex w_∞ is called the* sink vertex.
2. *The directed edge multiset $E_d(\mathcal{C}(\mathcal{T}))$ is partitioned into two sets, $E_1(\mathcal{C}(\mathcal{T}))$ and $E_2(\mathcal{C}(\mathcal{T}))$ constructed as follows. For each $w \in K(\mathcal{T})$ choose $v \in V(\mathcal{T})$ a representative of w. For each (v, v') edge in $E(\mathcal{T})$*
   (a) *if v' is in position w' then add a direct edge from w to w' to the multiset $E_1(\mathcal{C}(\mathcal{T}))$,*
   (b) *if v' is a leaf node then add a directed edge from w to w_∞ to the multiset $E_2(\mathcal{C}(\mathcal{T}))$.*

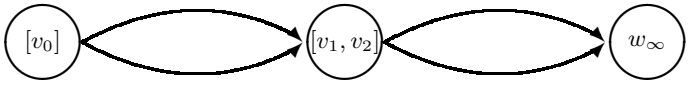

FIG 3. *CEG for Example 1*

3. *The set of undirected edges is*

$$E_u(\mathcal{C}(\mathcal{T})) = \{\{w, w'\} : \text{ with } w \neq w' \text{ and there exist } v, v' \in S(\mathcal{T}), u \in L(\mathcal{T}) \text{ with } v, v' \in u \text{ and } v \in w, v' \in w'\}$$

that is, undirected edges join positions in the same stage.

4. *If $e_1, e_2 \in E_d(\mathcal{C}(\mathcal{T}))$ and $\pi(e_1) = \pi(e_2)$ in the original tree, then e_1 and e_2 have the same colour in the CEG.*

By Item 4 above, the primitive probabilities give the colouring of the directed edges. Undirected edges are not coloured.

The CEG is as expressive as a tree for a sample space. However usually it has many less edges and its topology expresses a set of conditional independence statements. There are as many directed root-to-leave paths in the original tree as there are root-to-sink paths in its CEG. This is important because, in the network analogy, a single draw of a typical unit, whose process is described by the simulator network, corresponds to a root-to-sink path in its CEG.

If two vertices v and v^* of the original tree are in the same position, then for each path $\lambda(v, v_M)$ in the sub-tree $\mathcal{T}(v)$ there exists a corresponding path $\lambda^*(v^*, v^*_M)$ in $\mathcal{T}(v^*)$ along which the same evolutions occurs. This implies $\pi(\lambda) = \pi(\lambda^*)$. In particular consider the root-to-leaf paths, given in terms of vertices, $\lambda(v_0, \ldots, v, \ldots, v_M)$ and $\lambda^*(v_0, \ldots, v^*, \ldots, v^*_{M^*})$ where v_M and $v^*_{M^*}$ are leaves in $\mathcal{T}$ and v, v^* are in the same position. Then

$$\pi(\lambda) = \pi(\lambda(v_0, v))\pi(\lambda(v, v_M))$$
$$\pi(\lambda^*) = \pi(\lambda^*(v_0, v^*))\pi(\lambda(v, v_M))$$

Because the stage set is a refinement of the partition given by the position we can set $\Pi(\mathcal{C}) = \Pi(\mathcal{T})$ and $L(\mathcal{C}) = L(\mathcal{T})$.

Example 9 Figure 3 gives the CEG for the example in Figure 1 in the case of independence between X and Y that is when v_1 and v_2 are in the same stage.

Example 10 Figures 4 and 5 give a tree and its CEG for the stage set $\{\{v_0\}, \{v_1, v_3, v_{13}, v_{17}\}, \{v_2, v_7\}, \{v_5, v_9\}, \{v_{19}\}\}$ and the position set $\{\{v_0\}, \{v_1, v_3\}, \{v_5, v_9\}, \{v_2\}, \{v_7\}, \{v_{13}\}, \{v_{17}\}, \{v_{19}\}, w_\infty\}$.

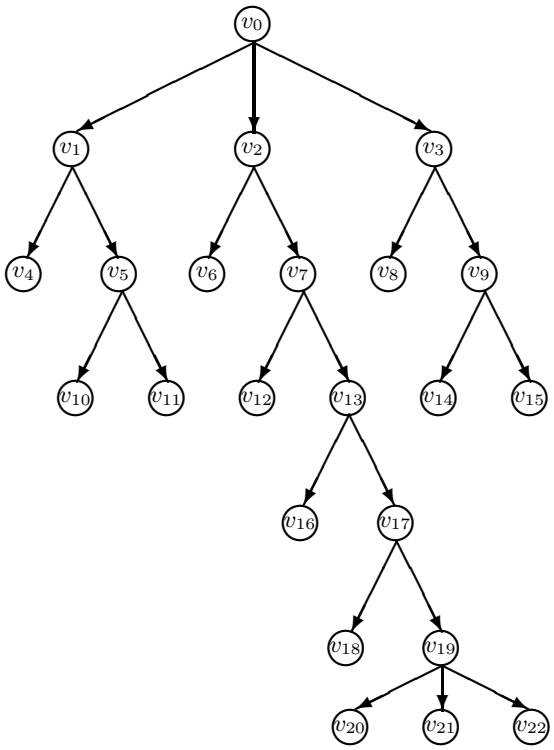

FIG 4. *A tree with unusual stage set*

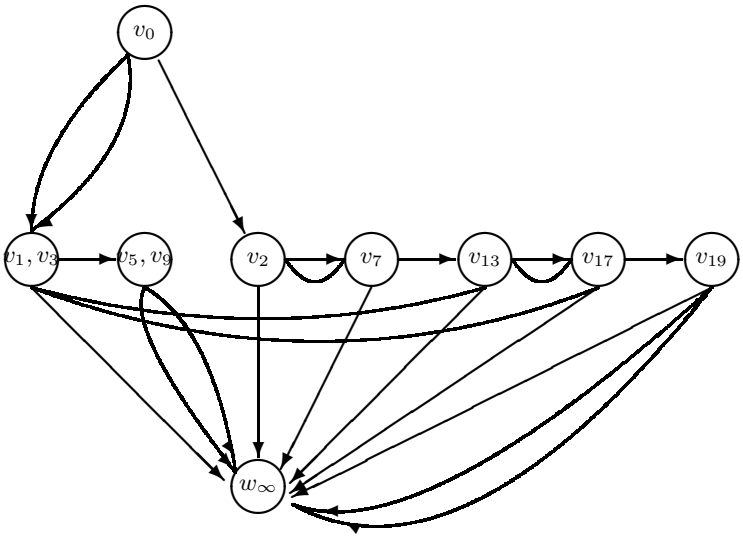

FIG 5. *CEG for the event tree in Figure 4*

Example 11 (Bayesian network) In [40] the authors prove that any discrete Bayesian network G on the random variables $\{X_1, \ldots, X_n\}$ can be fully expressed by a CEG $\mathcal{C}(G)$. Like context specific BNs [3] but unlike the probability decision graph [18] or the probability graph [4], the CEG provides a generalisation of the BN. Thus suppose X_i has parents $Q_i \subseteq \{X_1, \ldots, X_{i-1}\}$ in G, $2 \leq i \leq n$. Then two situations $v_{i-1} = (x_1, x_2, \ldots, x_{i-1})$ and $v'_{i-1} = (x'_1, x'_2, \ldots, x'_{i-1})$ are in the same stage u_i of $\mathcal{C}(G)$ if and only if the values of their parents agree. This fully expresses the conditional independence statement embodied in the BN.

Topological characteristics of a CEG derived from a discrete BN include: *(i.)* all the root-to-sink paths have the same length, *(ii.)* the stages consist of situations all of whose distances (length of the path from the root to the situation) from the root are the same, and *(iii.)* for $2 \leq i \leq n$ all stages u_i associated with different configurations of parents of X_i contain exactly the same number of situations. Various examples and d-separation theorems for CEGs are given in [40].

3. Conjugate estimation in chain event graphs

One appealing property of a BN is that under appropriate sampling regimes it supports a product Dirichlet multinomial conjugate analysis on its joint distribution, provided probabilities respect local and global independence: see for example [9, 17, 42]. In this section we show that this property is shared by CEGs.

To sidestep foundational issues associated with the appropriateness of the simulator network, consider the problem of how to estimate the primitive probabilities associated with each simulator/situation from a computer experiment. Here we assume that random and independent draws are taken from simulators lying along a path in $\mathcal{C}(\mathcal{T})$ that begins at the root vertex.

Run the tree simulator t times. Let $N[u]$ be the number of times we pass through a stage u so that there are $N[u]$ independent replicates $\{X_m(u) : 1 \leq m \leq N[u]\}$ from a random variable with the same distribution of $X(v)$ for a v in the stage u. Let $N_v[u]$ be the random variable counting the number of $X_m(u)$ taking value v for $v \in \mathbb{X}(u)$, where $\mathbb{X}(u)$ is the sample space of $X(v)$ for a v representative of u. This provides the basis for the construction of the likelihood function associated with this computer experiment.

A full Bayesian model on a CEG $\mathcal{C}$ is given by the triple $(\mathcal{C}, \Pi(\mathcal{C}), \mathcal{P})$ where $\mathcal{P}$ is the observer's distribution to be updated over time starting from $t = 0$.

Definition 12 *Let $\boldsymbol{\pi}(u) = (\pi(v|u) : v \in \mathbb{X}(u))$ be the vector whose components are the elements of $\Pi(u)$ and $\boldsymbol{\pi}(\mathcal{C})$ the analogous vector for $\Pi(\mathcal{C})$.*

Let $(\mathcal{C}, \Pi(\mathcal{C}), \mathcal{P})$ be a CEG with distribution $\mathcal{P}$ with density $p(\boldsymbol{\pi}(\mathcal{C}))$. The density $p(\boldsymbol{\pi}(\mathcal{C}))$ is called local *if*

$$p(\boldsymbol{\pi}(\mathcal{C})) = \prod_{u \in L(\mathcal{T})} p_u(\boldsymbol{\pi}(u))$$

where $p_u(\boldsymbol{\pi}(u))$ is a function only of its argument $\boldsymbol{\pi}(u)$, $u \in L(\mathcal{T})$.

Theorem 13 *If an observer's prior distribution on the CEG* $(\mathcal{C}, \Pi(\mathcal{C}), \mathcal{P}^{[0]})$ *is local then the posterior density* $p^{[t]}(\boldsymbol{\pi}(\mathcal{C}))$ *after* t *independent runs of the simulator from the root is also local so that*

$$p^{[t]}(\boldsymbol{\pi}(\mathcal{C})) = \prod_{u \in L(\mathcal{T})} p_u^{[t]}(\boldsymbol{\pi}(u))$$

Furthermore, if the components of the observer's prior densities $p_u^{[0]}(\boldsymbol{\pi}(u))$ *are all Dirichlet so that, on its simplex, in the notation above,*

$$p_u^{[0]}(\boldsymbol{\pi}(u)) \propto \prod_{v \in \mathbb{X}(u)} \pi(v|u)^{\alpha_{u,v}[0]-1}$$

(where $\alpha_{u,v}[0]$ *are the parameters of the Dirichlet distribution) then the components* $p_u^{[t]}(\boldsymbol{\pi}(u))$, $u \in L(\mathcal{T})$ *of the observer's posterior density are also Dirichlet densities given by*

$$p_u^{[t]}(\boldsymbol{\pi}(u)) \propto \prod_{v \in \mathbb{X}(u)} \pi(v|u)^{\alpha_{u,v}[t]-1}$$

where

$$\alpha_{u,v}[t] = \alpha_{u,v}[0] + n_v[u]$$

and $n_v[u]$ *is the observed value of the random variable* $N_v[u]$.

Proof. The form of the computer experiment provides a likelihood $L^{[t]}(\boldsymbol{\pi}(\mathcal{C}))$ of the probabilities of the outputs of the different simulators in the network from the t path simulations that separates over the probabilities in the stages so that

$$L^{[t]}(\boldsymbol{\pi}(\mathcal{C})) = \prod_{u \in L(\mathcal{C})} \prod_{v \in \mathbb{X}(u)} \pi(v|u)^{n_v[u]}$$

for each $u \in L(\mathcal{C})$ we have $\pi(v|u) > 0$ for all $v \in \mathbb{X}(u)$ and $\sum_{v \in \mathbb{X}(u)} \pi(v|u) = 1$.

Now Bayes rule gives us immediately that

$$p^{[t]}(\boldsymbol{\pi}(\mathcal{C})) \propto \prod_{u \in L(\mathcal{T})} p_u^{[t]}(\boldsymbol{\pi}(u))$$

Since $p^{[t]}(\boldsymbol{\pi}(\mathcal{C}))$ must integrate to unity this proportionality must in fact be an equality. Under the prior Dirichlet hypothesis the posterior density clearly retains the Dirichlet monomial form in probabilities with the new powers of the monomial term $\pi(v|u)$ given by $\alpha_{u,v}[t] - 1$. This completes the proof. ■

This provides the obvious analogue to the conjugate prior to posterior analysis for a BN given ancestral data as outlined in, for example, [42]. Complete data sampling is a special case of the above where all sampled paths are root-to-sink paths in $\mathcal{C}$. In the subclass of CEG models constituting discrete BNs it is now well known that Dirichlet conjugacy is lost when sampling is non-ancestral, even in the simplest models. Indeed it is often the case that

certain functions of the parameters are unidentifiable when certain interior positions are unobserved. This is because it is then only possible to learn directly about collections of polynomials in primitives and these may or may not have unique solutions see, for example, [38]. The CEG, coding as it does more asymmetric collections of monomials, suffers from the same difficulty. The observer then sees the result of a simulator whose stage – and hence its associated component simulator – is uncertain to her. So except in certain degenerate circumstances, conjugacy and sometimes identifiability is lost under non-ancestral sampling.

Assuming a local prior and that t independent path draws beginning at the root vertex are drawn, the predictive mass function $q^{(t)}_{t+1}(\lambda_{t+1})$ of a subsequent root-to-sink path draw from the simulator network, represented by $\lambda_{t+1} = \lambda$, is given by

$$\begin{aligned} q^{(t)}_{t+1}(\lambda) &= \int \pi(\lambda) p^{[t]}(\boldsymbol{\pi}(\mathcal{C}))\, d\boldsymbol{\pi}(\mathcal{C}) \\ &= \int \prod_{j=0}^{n[\lambda]} \pi(w_{j+1}|w_{j,\lambda}(u_j)) p^{[t]}(\boldsymbol{\pi}(\mathcal{C}))\, d\boldsymbol{\pi}(\mathcal{C}) \end{aligned}$$

where, in $\mathcal{C}$,

$$\lambda = \lambda(w_0, w_1, ..., , w_{n[\lambda]})$$

Under the local Dirichlet prior given above this is the expectation of a monomial in the probabilities of a product Dirichlet. So it can be expressed explicitly — albeit via a rather complicated formula — as the product and quotient of gamma functions whose parameters are linear functions of the posterior hyper-parameters defining the posterior density $p^{[t]}(\boldsymbol{\pi}(\mathcal{C}))$. It is easily checked that any polynomial functions of future draws can also be found explicitly as an even more complicated function of gamma functions. An important special case is the predictive distribution of the next N draws.

In general, if each of the positions $\{w_0, w_1, ..., , w_{n[\lambda]}\}$ along the path λ lie in distinct stages $\{w_k \in u_k : 0 \leq k \leq n[\lambda]\}$ — as they do for example in a BN — then a local prior, $q^{(t)}_{t+1}(\lambda)$ will also take a product form. Explicitly

$$q^{(t)}_{t+1}(\lambda) = \prod_{k=1}^{n[\lambda]} q^{(t)}_{t+1,k}(\boldsymbol{\pi}_k(u_k))$$

where, for $0 \leq k \leq n[\lambda]$, if u_k is the k numbered stage, so that $\boldsymbol{\pi}_k(u_k) = \pi(v(k)|u)$ where $v[k]$ is an index of the child of u_k in λ, then

$$\begin{aligned} q^{(t)}_{t+1,u}(\boldsymbol{\pi}_u(u_\lambda)) &= q^{(t)}_{t+1,u}(\pi_{v(k)}[u]) \\ &= \int \pi_{v(k)}[u] p^{[t]}_u(\boldsymbol{\pi}(u))\, d\boldsymbol{\pi}(u) \end{aligned}$$

Note that predictive random variables $\{X_{t+1}^{(t)}(u) : u \in L(\mathcal{C})\}$ whose distributions are given by

$$P(X_{t+1}^{(t)}(u) = v) = \pi_v^{(t)}[u] = \int \pi_v[u] p_u^{[t]}(\boldsymbol{\pi}(u))\, d\boldsymbol{\pi}(u)$$

$v \in \mathbb{X}(u)$, provide the obvious predictive analogue to $\{X(u) : u \in L(\mathcal{C})\}$. In particular, if the observer has a local prior, the mutual independence of $\{X(u) : u \in L(\mathcal{C})\}$ will imply the mutual independence of $\{X_{t+1}^{(t)}(u) : u \in L(\mathcal{C})\}$.

Note that when the Dirichlet product conjugate prior is appropriate, from the above we have that

$$\begin{aligned} q_{t+1,u(k)}^{(t)}(\boldsymbol{\pi}(u_k)) &= \int \pi(v(k)|u) \prod_{v \in \mathbb{X}(u)} \pi(v|u)^{\alpha_{u,v}[t]-1}\, d\boldsymbol{\pi}(u) \\ &= \alpha_{u,v(k))}[t](\overline{\alpha}_{u,v(k)}[t])^{-1} \end{aligned}$$

where

$$\overline{\alpha}_{u,v(k)}[t] = \sum_{v \in \mathbb{X}(u)} \alpha_{u,v}[t]$$

Let $\Pi^{(t)}(\mathcal{C})$ be the set of predictive probabilities after t simulations. Suppose that the CEG $(\mathcal{C}, \Pi(\mathcal{C}))$ accurately expresses the observer's beliefs. It follows that if her prior beliefs are local then, after observing t observations, her beliefs about the next observation are described by the CEG $(\mathcal{C}, \Pi^{(t)}(\mathcal{C})))$, with the same graphical topology where $\Pi^{(t)}(\mathcal{C})$ defined above substitutes "best estimates" for their corresponding true values of $\Pi(\mathcal{C})$. Therefore, from a Bayesian perspective, the study of dependence structures when primitive probabilities are known gives valuable insight into predictive dependence structures and applies directly to analogous statements at time t, albeit only when beliefs are local. Thus for clarity the remainder of this paper, we assume all primitive probabilities are known. Analogous constructions and results can then be applied to estimated systems using the correspondence above.

3.1. Model selection

Next, we briefly address how to perform model selection on a class of CEGs. Thus suppose the observer's beliefs are accurately expressed by one of the CEGs in a class $\mathbb{C}$ where the prior probability that CEG $\mathcal{C}^{(c)} \in \mathbb{C}$ is the right model is $\mathbb{P}^{(c)}[0]$. A popular method of Bayesian model selection chooses a model with the highest log posterior probability [5, 11]. Here the CEG $\mathcal{C}^{(c)}$ marginal likelihood density $q_t^{(t)}(\lambda)$ of what we observe given $\mathcal{C}^{(c)}$ is

$$\log q_t^{(t)}(\lambda) = \sum_{k=1}^{n[\lambda]} q_{t,u}^{(t)}(\pi_{v(k)}(u))$$

where

$$q_{t,u}^{(t)}(\pi_{v(k)}(u)) = \sum_{v\in\mathbb{X}(u,\mathcal{C}^{(c)})} \{\log\Gamma(\alpha_{u,v}^{(c)}[t]) - \log\Gamma(\alpha_{u,v}^{(c)}[0])\} - \log\Gamma(\sum_{v\in\mathbb{X}(u,\mathcal{C}^{(c)})} \alpha_{u,v}^{(c)}[t]) + \log\Gamma(\sum_{v\in\mathbb{X}(u,\mathcal{C}^{(c)})} \alpha_{u,v}^{(c)}[0]) + \log\Gamma(\sum_{v\in\mathbb{X}(u,\mathcal{C}^{(c)})} n_v[u]+1) - \sum_{v\in\mathbb{X}(u,\mathcal{C}^{(c)})} \log\Gamma(n_v[u]+1)$$

where Γ is the Gamma function and

$$a_{u,v}^{(c)}[t] = a_{u,v}^{(c)}[0] + n_v[u]$$

Relative to an arbitrary reference CEG $\mathcal{C}^{(0)}$ the posterior probability $\mathbb{P}^{(c)}[t]$ is uniquely calculable from the familiar log-odds formula

$$\begin{aligned}\sigma^{(c)} &= \log\mathbb{P}^{(c)}[t] - \log\mathbb{P}^{(0)}[t] \\ &= \left\{\log\mathbb{P}^{(c)}[t] - \log\mathbb{P}^{(0)}[t]\right\} + \left\{q_{t,u}^{(t)}(\pi_{v(k)}(u)) - q_{t,u}^{(0)}(\pi_{v(k)}(u))\right\}\end{aligned}$$

Thus $\sigma^{(c)}$ acts as a score function: by choosing the CEG $\mathcal{C}^{(c)}$ with the highest score we choose the model with the highest posterior probability. Clearly setting the prior probabilities to all CEGs considered equal simplifies this score, removing its dependence on $\mathbb{P}^{(c)}[0]$.

Of course, as with the analogous problem for selection of BNs using this method we still have the knotty problem of specifying appropriate priors for the different CEGs in $\mathbb{C}$. Although a discussion of this point is outside the scope of this paper we note that it is straightforward to use techniques evoking the principles that beliefs about different models should correspond to the same "dummy sample" [21] or that models that are statistically equivalent are given the same prior probability [6].

4. Manipulation and Causality

4.1. Manipulations

Like a BN, a CEG provides a flexible framework for expressing what might happen were a model manipulated in certain ways. The validity of such a framework is of course heavily dependent on context. Shafer [36] argues similarly for probability trees. Through developing notions of causality in terms of simultaneous equation models implicitly Pearl [27] uses the context of simulator networks in his definition of causal effects.

This context is extremely valuable because it defines a domain where the predicted effects of a proposed manipulation is uncontentious. Here we follow Pearl; first developing a model for the manipulation of a network of simulators and then leaving the issue of whether a manipulation of a real context has an effect exactly analogous to the manipulation of a simulator network, to

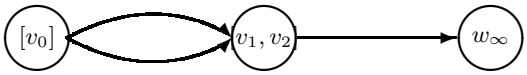

FIG 6. *Manipulated CEG for Example 15*

practical considerations particular of the application under study. Translated into our framework therefore, manipulation of a set of situations $D \subset S(\mathcal{T})$ corresponds to substituting the simulator whose output is governed by $X(v)$, $v \in D$, by another simulator whose output is governed by the random variable $X(v)$ with a different probability mass function $\widehat{\pi}(\cdot|v)$, $v \in D$. Having made this substitution, the network is then run as before. Note that the type of atomic manipulation that provides the focus of Pearl's work [27, 28] sets $\{X(v) : v \in D\}$ so that $\widehat{\pi}(\cdot|v)$ is a degenerate probability mass function with all its probability mass on a preassigned child of v in $\mathcal{T}$.

Some discussions of notions of the manipulation of a system and intervention and various applications can be found in [15, 27, 36, 43].

Definition 14 *Let $(\mathcal{T}, \Pi(\mathcal{T}))$ be a probability tree model and $D \subset S(\mathcal{T})$ a subset of situations. A* manipulation *on D of the tree is a triple $(D, (X(v) : v \in D), (\widehat{\Pi}(v|D) : v \in D))$ where $\widehat{P}(X(v) = v') = \widehat{\pi}(v'|v)$ for $v' \in \mathbb{X}(v)$ is a distribution of $X(v)$ and $\widehat{\Pi}(v|D) = \{\widehat{\pi}(v'|v) : v' \in \mathbb{X}(v)\}$.*

The effect *of this manipulation is the transformation $(\mathcal{T}, \Pi(\mathcal{T})) \to (\mathcal{T}, \widehat{\Pi}(D))$ where*

$$\widehat{P}(X(v) = v') = \begin{cases} \pi(v'|v) & \text{if } v \notin D \\ \widehat{\pi}(v'|v) & \text{if } v \in D \end{cases}$$

for $v' \in \mathbb{X}(v)$. The manipulated tree *is the probability tree model so obtained. The* manipulated CEG *is the CEG of the manipulated tree.*

If two unmanipulated situations were in the same stage of the original tree and are manipulated in the same way, then they remain in the same stage in the manipulated tree.

It seems to us that Definition 14 is the obvious choice of definition of the (effect of) manipulation when dealing with simulator networks. It may not be reasonable in other cases, see for example Shafer [36, Section 4.5]. It will always be necessary to check whether the simulator analogy extends to a given context.

Example 15 (cont. Example 4) Let v_1 and v_2 be in the same stage with v_3 mapping into v_5. Let $D = \{v_1, v_2\}$ and $\widehat{P}(X(v_1) = v_3) = 1$, $\widehat{P}(X(v_2) = v_5) = 1$. The CEG of the manipulated tree is in Figure 6 where we did not draw the edge out of the "manipulated position" and into w_∞ with zero probability. Later we shall come back to the idea of a manipulated position.

A probability tree model $(\mathcal{T}, \Pi(\mathcal{T}))$ is said to be *valid for an application* if its associated simulation network accurately expresses the observer's beliefs

about how situations happen in that application. A CEG model $(\mathcal{C}, \Pi(\mathcal{C}))$ is *valid for an application* if so is its underlying probability tree.

A manipulated tree can be valid for an application even when its idle version is not. An important example of a valid tree is the decision tree $\mathcal{T}$ where $D \subset S(\mathcal{T})$ is the set of decision nodes. The unfolding of situations corresponding to each (non-randomised) decision rule in the decision tree is given by the tree itself. The primitive probabilities in $\widehat{\Pi}(D)$, not associated with the degenerate distributions associated with the chosen manipulation/decision rule, are simply the probabilities associated with the chance nodes $S(\mathcal{T}) \setminus D$. Clearly in this context sets of situations labelling decision nodes in $\mathcal{T}$ have degenerate distributions.

Furthermore, the manipulation of a real system (for example, by controlling values of covariates, employing randomised designs and so on) is a common way of trying to ensure that a simulator network analogy of the manipulated system might be at least plausible when the simulator network analogy to the idle system certainly would *not* be valid: for an example see [22, 23]. This is extremely important because estimated parameters may be parameters associated with the effects after a planned manipulation/treatment regime where no idle system currently exists.

4.2. Causal probability trees

In any context there must be a practical manipulation of the real system we hypothesize corresponds to our actual manipulation. For any given application modelled by a simulator network there may be ways in which we envisage performing a manipulation in practice. In a medical context, for example, such a manipulation might be a certain type of treatment regime e.g. "whenever a unit lies in a position w treat it so that it always moves to a position w'" where w and w' are connected by a direct edge. Note that for a valid CEG such a manipulation will be well defined. The issue is then whether the observer believes that the corresponding manipulation of the simulator network faithfully describes her beliefs about the relationship between the variables under her chosen real manipulation.

Analogously to Pearl [27, Definition 1.3.1], we say that a probability tree model $(\mathcal{T}, \Pi(\mathcal{T}))$ is *causal* (written a CPT) if for any manipulation the manipulated tree $(\mathcal{T}, \widehat{\Pi}(D))$ is valid.

Note that the probability tree model of an application will be causal if and only if any appropriately defined real world manipulation has the same effect as the manipulation of the corresponding simulator network. Moreover note that as pointed out by Shafer [36] a manipulation could destroy the inherent conditional independence statements, here expressed through the position and stage partitions in the unmanipulated system.

There are important classes of manipulations associated with a CEG.

Definition 16 *A manipulation is called* positioned *if the partition of positions after the manipulation is equal to or a coarsening of the partition before manipulation. It is called* staged *if the partition of stages after the manipula-*

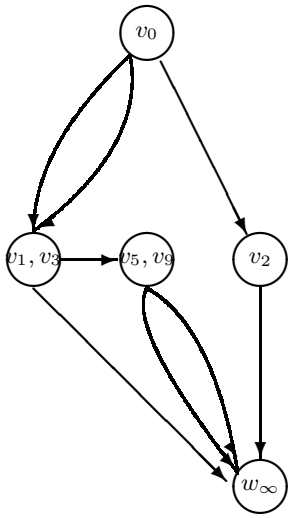

FIG 7. *Manipulated CEG for Example 17*

tion is equal to or a coarsening of the partition before manipulation.

Example 17 Example 15 is an example of a staged manipulation and trivially also of a positioned manipulation.

Example 18 In Figure 4 the staged manipulation defined by

$$\widehat{\pi}(v_5|v_1) = 1, \quad \widehat{\pi}(v_9|v_3) = 1, \quad \widehat{\pi}(v_{17}|v_{13}) = 1, \quad \widehat{\pi}(v_{19}|v_{17}) = 1$$

will lead to a CEG like that in Figure 5 in which the edges into w_∞ from the positions $[v_1, v_3]$, $[v_{13}]$ and $[v_{17}]$ are removed as the associated manipulated probabilities become zero.

The staged manipulation corresponding to $\widehat{\pi}(v_6|v_2) = 1$ and $\widehat{\pi}(v_{12}|v_{17}) = 1$ cuts off the branch starting at v_2 through v_7 from the probability tree in Figure 4 again because the probability of passing through v_7 is zero. The CEG of the manipulated tree is in Figure 7.

A positioned manipulation manipulates all sample units identically when their future development distributions are identical, using the same (possibly randomising) allocation rule. A staged manipulation will treat sample units identically if their next development in the idle system is the same. In our experience in practice it seems often appropriate to restrict study to positioned manipulations. All manipulations on a BN considered by Pearl are also necessarily staged. Example 19 gives a simple case when a staged manipulation is not reasonable.

Example 19 An English university has residence blocks of flats with two rooms each. It allocates prospective second year students (either English (E) or Chinese (C)) to one of the two rooms of each flat. The second room has to be allocated to a prospective first year student. In the past this has been

done at random. However it has been noticed in a survey that the probability of satisfaction of home students placed with home students is higher and of Chinese students placed with Chinese students is higher than when they are mixed. In order to cause students' satisfaction to increase, the university decides to place first year students with a second year student with the same ethnicity.

The BN and CEG of this problem are given in Figure 8 where X represents the ethnicity of the second year student, Y that of the first year student and Z is a binary index of the satisfaction of two students in the same flat, taking values U and S. Thus for example $X(v_0) = X$, $X(v_1) = [Y|X = E]$, $X(v_3) = [Z|X = E, Y = E]$ and $\pi(v_5|v_2)$ gives the probability of allocating a Chinese first year student to a flat with a Chinese second year student. The vertices v_3 and v_5 are in the same stage to indicate a non-mixed flat, analogously interpretation has the stage v_4, v_6. The undirected edge between v_1 and v_2 represents the random allocation of the first year student to a flat.

The relationship between satisfaction and shared race is not depicted in the BN whilst it is in the CEG through the colouring of its edges. More significantly it is impossible to determine, either from the semantics of the BN or the factorisation of the probability mass function of the path events, whether the allocation of the prospective second year student occurs before the allocation of the prospective first year student. The CEG states that second year allocation occurs before first year allocation explicitly, so that "causal" manipulation of the type suggest by the survey above is a possibility. The semantic of a BN is not refined enough to represent the sort of quite legitimate manipulation considered in this example.

A manipulation that forces individuals of the same ethnicity to share a flat implies a CEG without the direct edge between v_1 and v_2 and without the crossing arrows in the CEG of Figure 8.

4.3. Manipulating CEGs

The CEG can be used as a framework for positioned manipulations. So it lies usefully between the transparent but restrictive class of models fully expressed by a causal BN and the CPT which is extremely expressive but rather too demanding for many purposes, because a CPT requires that any manipulated tree is valid.

Definition 20 *A collection W of positions of a CEG is called a* fine cut *if all paths from w_0 to w_∞ pass through exactly one element of W.*

In particular $W = \{w_0\}$ is a fine cut. In the CEG of Figure 5 the set $\{[v_1, v_3], [v_2]\}$ is a fine cut of minimal size.

Just as in the causal BN it is possible to prove various results about identifiability of an effect of a staged manipulation simply from the topology of the CEG. The next section concerns inferences that can be made about a manipulated simulator network from observing certain statistics of random samples taken from the corresponding idle network.

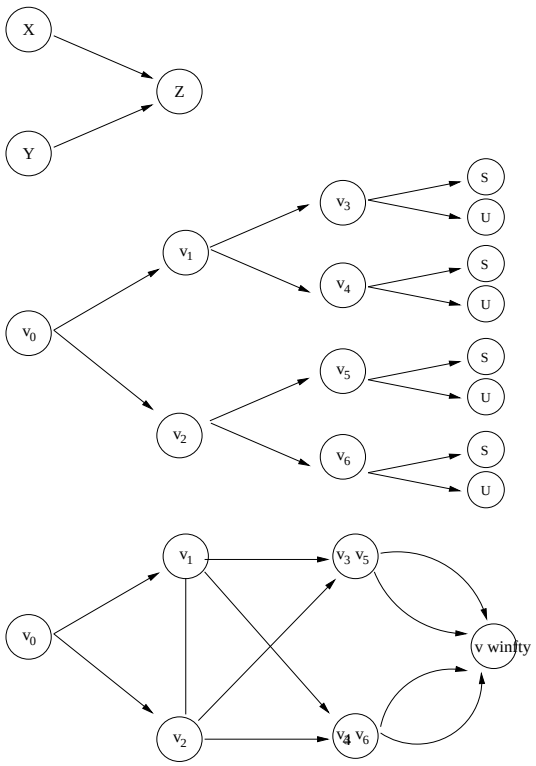

FIG 8. *BN and CEG for the example of university room allocation*

The standard manipulations of a BN are those that force the outputs of some of the components of a simulator network to take pre-assigned values [27, 28]. The analogue for the CEG is to consider manipulations which force all the paths of a simulator network to pass through an identified set of positions W. For example the assignment of a particular type of unit, here described by their current position, to a particular treatment regime, here described by a set of subsequent positions W. A set of positions W fulfills the role described above and labels a manipulation unambiguously if it exhibit certain properties.

For a CEG $\mathcal{C}$ and a set of position W in $\mathcal{C}$, let $\mathrm{pa}(W)$ denote the set of all parents of the elements in W, that is $\mathrm{pa}(W) = \{w^* \in V(\mathcal{C}) : \text{there exists } w \in W \text{ such that } (w^*, w) \in E_d(\mathcal{C})\}$.

Definition 21 *A subset W of positions of a CEG $\mathcal{C}$ is called a* manipulation set *if*

1. *all root-to-sink paths in $\mathcal{C}$ pass through exactly one position in* $\mathrm{pa}(W)$*, and*
2. *each position in* $\mathrm{pa}(W)$ *has exactly one child in W.*

Example 22 In the CEG in Figure 6 the position $[v_1, v_2]$ is a manipulation set. Excluding the trivial case of a manipulation set consisting of the root node only, there is no manipulation set in the CEG in Figure 5.

In the analogy above the set $\mathrm{pa}(W)$ will then correspond of the positions any unit must reach to be submitted to a treatment forcing them into the position W. Note that when W is a manipulation set then all units will be submitted to the treatment regime. Although the CEG obviously does not force us to consider only manipulation to a manipulation set – indeed many manipulations we might like to consider may be more general than this – it is straightforward to develop theorems about such manipulations analogous to those for causal BNs. Note, in particular, that all manipulations considered for causal BNs are to a manipulation set as is the type of manipulation described in Example 19.

Definition 23 *A manipulation $(D, (X(v), v \in D), (\widehat{\Pi}(v|D) : v \in D))$ of a complete CEG is called a* pure manipulation to the positions W *if*

1. *it is a positioned manipulation,*
2. *for each $v \in D$ there exists $w \in W$ such that $\widehat{\mathrm{P}}(X(v|D) = w) = \widehat{\pi}(w|v) = 1$ and*
3. *no $v \notin D$ is manipulated.*

Definition 24 *A CEG $(\mathcal{C}, \Pi(\mathcal{C}))$ is called* causal *if*

1. *$(\mathcal{C}, \Pi(\mathcal{C}))$ is valid and*
2. *for the pure manipulation to any manipulation set of $\mathcal{C}$ the corresponding manipulated CEG is also valid.*

If a CEG admits a description as a BN and the CEG is causal then so is the BN. So in this sense a causal CEG is a natural generalisation of the

causal BN, applicable to asymmetric models. It is however amenable to more varied types of manipulations: for example those based on certain functions of preceding variables as in Example 19.

5. Identifying effects of a manipulation

5.1. Identification of causal effects

Considerable recent interest in causal BN literature is in studying when the effect of a manipulation on a prespecified random variable Y can be identified from observing a subset of its variables that are observed or "manifest". Typically sufficient conditions on the topology of the BN are given for such identifiability to exist. This allows us to design experiments on the original "idle" system so as to be able to estimate effects on a manipulated system: for example the effects of a proposed new treatment regime. The topology of the CEG can also be used for this purpose. Indeed it can be used to find *functions* of the data (not just subsets of possible measurements) that when observed in the idle system allows allows us to estimate all the effects of a given manipulation of a causal CEG. As in [27] we prove several sufficient conditions for identifiability and generalise Pearl's Backdoor theorem to CEG models.

We first need some definitions. We begin by stating what it means for a random variable, measurable with respect to the path σ-algebra of a CEG, to be observed or manifest.

5.1.1. Manifest random vectors

To a path $\lambda(v_1, v_2)$ between the vertices v_1 and v_2 in a probability tree corresponds a path $\lambda(w_1, w_2)$ in the CEG between two positions. $\lambda(w, w_\infty)$ is a path through w and the sink node. Thus the path σ-algebra on the probability tree maps into a σ-algebra on the CEG. However the paths between two positions are easier to specify on a CEG than on an event tree because they correspond to the set of paths between two vertices of the CEG itself. Note that in the CEG there may be more than one path between two positions.

A random vector $\mathbf{M}$ is $\mathcal{C}$-measurable if it is measurable with respect to the path σ-algebra induced on the CEG. Let $\Omega_{\mathbf{M}}$ denote the sample space of $\mathbf{M}$. For each value $\mathbf{m} \in \Omega_{\mathbf{M}}$, let $\Lambda_{\mathbf{m}}$ be the set of paths corresponding to the event $\{\mathbf{M} = \mathbf{m}\}$. Let $\Lambda_{\mathbf{m}}(w_1, w_2)$ denote the set of paths in the CEG that pass through first w_1 and next w_2, and are contained in the event $\{\mathbf{M} = \mathbf{m}\}$. If W_1 and W_2 are fine cuts, then

$$\{\Lambda_{\mathbf{m}}(w_1, w_2) : w_1 \in W_1, w_2 \in W_2, \mathbf{m} \in \Omega_{\mathbf{M}}\}$$

forms a partition of the set of paths in the CEG. Other partitions are given by $\{\Lambda_{\mathbf{m}}(w_0, w) : \mathbf{m} \in \Omega_{\mathbf{M}}\}$ and $\{\Lambda_{\mathbf{m}}(w, w_\infty) : \mathbf{m} \in \Omega_{\mathbf{M}}\}$ for any position w.

Definition 25 *A random vector* $\mathbf{M}$ *is called* observed *(or* manifest*) if and only if indicators on events in the path* σ*-algebra corresponding to the set of paths* $\Lambda_{\mathbf{m}}$ *is observed for all* $\mathbf{m} \in \Omega_{\mathbf{M}}$.

Definition 26 *Call a manipulation of a CEG* $(\mathcal{C}, \Pi(\mathcal{C}))$ forced to (the position) w *if*

1. *it assigns probability one to the event* $\{w\} = \{\lambda \in \mathbb{X} : w \in \lambda\}$, *and*
2. *all primitive probabilities in the manipulated CEG,* $\widehat{\Pi}(\mathcal{C})$, *associated with positions at or after* w *in* $\mathcal{C}$ *are those of the idle system.*

This means that a manipulation forced to w forces the network to pass through w by manipulations of the network before w but subsequently allows the path evolution to be governed by the original network of simulators. Note that when $\{w\}$ is a manipulation set of $\mathcal{C}$ the pure manipulation to w is a particular example of a manipulation forced to w.

Example 27 In Example 19 a manipulation forced to $w = [v_3, v_5]$ is obtained by setting $\widehat{\pi}(v_4|v_1) = 0 = \widehat{\pi}(v_6|v_2)$, that is by allocating students with the same ethnicity to the same flat. This manipulation directly on the CEG is given by $\widehat{\pi}([v_4, v_6]||[v_1]) = \widehat{\pi}([v_3, v_5]||[v_2]) = 0$.

First we prove some results for a manipulation forced to a position w and next extend them to a more general type of manipulation.

There is a natural domain for describing what happens after a manipulation forced to w. Let $(\mathcal{C}(w), \Pi(\mathcal{C}(w)))$ denote the CEG whose graph consists of all vertices and edges that lie on paths from w to w_∞ in the CEG and whose stages and their corresponding random variables are inherited from those of the original CEG. Let us call $\mathcal{C}(w)$ the *subCEG* of $\mathcal{C}$ forced to w. Note that the CEG $\mathcal{C}(w)$ describes what happens in the network $\mathcal{C}$ *after any* manipulation forced to w. Explicitly any atomic event associated with a root-to-sink path in $\mathcal{C}(w)$ is associated with a w-to-sink path in $\mathcal{C}$ and has an associated probability as the product of primitives in the original CEG – i.e. associated with the path through the network of simulators in $\mathcal{C}$ forced to w.

5.1.2. Effect random variable

To keep the analogy with the work by Pearl and others on identifiability, next we consider an effect random variable. We partition the set of root-to-sink paths in $\mathcal{C}(w)$ as $\Lambda_y^+(w, w_\infty)$ for y in some index set Ω_Y and interpret it as the sample space of a random variable $\widehat{Y}(w)$ on the manipulated CEG. The random variable $\widehat{Y}(w)$ could be a measurement of an effect after any manipulation forced to w.

There is a natural random variable, defined on the unmanipulated CEG, that can be associated with $\widehat{Y}(w)$: namely the one whose event $\{Y(w) = y\}$ consists of all paths in $\mathcal{C}$ passing through w and then continuing along a path in $\Lambda_y^+(w, w_\infty)$. Thus formally let $Y(w)$ denote a $\mathcal{C}$-measurable random variable such that $\{Y(w) = y\}$ if and only if $\lambda \in \Lambda_y(w_0, w)$. Observe that $\Lambda_y(w_0, w) = \Lambda^-(w_0, w) \times \Lambda_y(w, w_\infty)$ and $\Lambda^-(w_0, w)$ is the set of all truncated root-to-w paths in the unmanipulated CEG $\mathcal{C}$ and $\times$ indicates the concatenation of paths to give paths in $\mathcal{C}$.

Lemma 28 equates the probability of the event $\{Y(w) = y|w\}$ in the idle CEG with the probability of the event $\{\widehat{Y}(w) = y\}$ in the manipulated CEG.

Lemma 28 *For all $y \in \Omega_Y$, under a manipulation forced to w*

$$\widehat{P}(\widehat{Y}(w) = y) = P(Y(w) = y|w)$$

provided that in the unmanipulated system $P(w) > 0$.

Proof. By definition

$$\widehat{P}(\widehat{Y}(w) = y) = \widehat{\pi}(\Lambda_y^+(w, w_\infty)) = \pi(\Lambda_y^+(w, w_\infty))$$

where $\widehat{\pi}(\Lambda_y^+(w, w_\infty))$ denotes the probability that a root-to-sink path in $\mathcal{C}(w)$ will lie in $\Lambda_y^+(w, w_\infty)$. The last equality holds because the manipulation is forced to $\{w\}$. Directly from the construction of $(\mathcal{C}(w), \Pi(\mathcal{C}(w)))$ from $(\mathcal{C}, \Pi(\mathcal{C}))$ given above, we have

$$P(Y(w) = y, w) = \pi(\Lambda_y^+(w, w_\infty))\, \pi(\Lambda^-(w_0, w))$$

where $\pi(\Lambda^-(w_0, w))$ is the probability that a root-to-sink path in $\mathcal{C}$ will pass through w in the unmanipulated network. Since, for $\Lambda(w_0, w)$, the set of path passing through w in $\mathcal{C}$, we have $\pi(\Lambda^-(w_0, w)) = \pi(\Lambda(w_0, w))$, the result now follows from the definition of conditional probability. ■

So for a manipulation forced to w it is possible to observe indicators on the events $\{\Lambda_y(w_0, w) : y \in \Omega_Y\}$ in the unmanipulated system and to identify the effects on $Y(w)$ of the manipulation, using Lemma 28. An important special case of this occurs when a manipulation to a set W is a manipulation to a position w occurring after all positions in W: a graphical property of the CEG that can be easily identified by eye. So if such a manipulation is valid for a given application, then the effect on $Y(w)$ of the manipulation can be directly observed from the unmanipulated system. The formula in Lemma 28 is satisfied if we can find any position w such that, after enacting a manipulation, all paths pass through w in $\mathcal{C}$ and we can learn that the event $\{w\}$ occurs from our set of measurements.

It is not always possible, even in models that can be described by a causal BN, to observe indicators on the events $\{\Lambda(y, w) : y \in \Omega_Y\}$ for a suitable choice of w but only a set of coarser events. Nevertheless being able to observe indicators on the events $\{\Lambda(y, W) : y \in \Omega_Y\}$ where, for some W, for each $y \in \Omega_Y$,

$$\Lambda(y, W) = \bigcup_{w \in W} \Lambda(y, w)$$

can also be sufficient for identifiability. However to show this is less straightforward and first we need some further definitions.

Definition 29 *A set of positions W of a CEG $\mathcal{C}$ is called* $\mathcal{C}$-regular *if no two positions in W lie on the same directed path of $\mathcal{C}$.*

By definition, a manipulation set of $\mathcal{C}$ is always $\mathcal{C}$-regular.

Let W be a $\mathcal{C}$-regular set of positions. Define a new CEG, $(\mathcal{C}(W), \Pi(\mathcal{C}(W)))$ formed by connecting the subCEGs $\mathcal{C}(w)$, $w \in W$, to a new root vertex w_0^*

and retaining all edges between the $\mathcal{C}(w)$'s in the original CEG. For $w \in W$ the new edge (w_0^*, w) is labeled

$$P(X(w_0^*) = w) = \frac{\pi(\Lambda^-(w_0, w))}{\pi(\Lambda^-(w_0, W))}$$

where $\pi(\Lambda^-(w_0, w)) = \sum_{\lambda \in \mathbb{X}: w \in \lambda} \pi(\lambda)$ is the probability of passing through w in the original CEG and $\pi(\Lambda^-(w_0, W) = \sum_{w \in W} \sum_{\lambda \in \mathbb{X}: w \in \lambda} \pi(\lambda)$ is the probability of passing through a position in the set W in the original CEG. Note that because W is C-regular,

$$\sum_{w \in W} P(X(w_0^*) = w) = 1$$

Let $\{\Lambda_y^+(W, w_\infty) : y \in \Omega_Y\}$ denote any partition of the set of root-to-sink paths of $\mathcal{C}(W)$ where Ω_Y is an index set, for example Ω_Y is the set mentioned just before Definition 29. One way to construct a sample from $\mathcal{C}(W)$ using a sample from $\mathcal{C}$ is to simply reject all samples whose root-to-sink paths do not pass through W and accepting all others.

Let $Y(W)$ denote a $\mathcal{C}$-measurable random variable such that

$$\{Y(W) = y\} \Leftrightarrow \lambda \in \Lambda_y(w_0, W)$$

with

$$\Lambda_y(w_0, W) = \Lambda^-(w_0, W) \times \Lambda_y^+(W, w_\infty)$$

In this sense $Y(W)$ is a random variable that happens after W in the unmanipulated system.

Next we construct an effect random variable associated with a manipulation forced to W where W is a $\mathcal{C}$-regular set. $\widehat{Y}(W)$ denote a $\mathcal{C}(W)$-measurable random variable representing the effect if and only if for $y \in \Omega_Y$

$$\{\widehat{Y}(W) = y\} = \Lambda_y^+(W, w_\infty) \tag{2}$$

so that Ω_Y is the sample space of $\widehat{Y}(W)$.

Next we look for sufficient conditions on the manipulation and on the topology of the unmanipulated CEG so that $w \in W$

$$\widehat{P}(X(w_0^*) = w) = \frac{\pi(\Lambda^-(w_0, w))}{\pi(\Lambda^-(w_0, W))} = P(X(w_0^*) = w) \tag{3}$$

holds. That is, we want that an effect of a manipulation forced to a $\mathcal{C}$-regular set of positions W can be determined directly from probabilities in the unmanipulated system. We do this through the notion of an amenable manipulation. We need to construct a graph representing what happens until we reach a given position w. Let $\mathcal{C}^*(w)$ denote the coloured subgraph of $\mathcal{C}$ whose vertices and edges are those along the root-to-w paths in $\mathcal{C}$ and whose edge colouring is inherited from $\mathcal{C}$ as well. Usually $\mathcal{C}^*(w)$ is not a CEG. Write

$K(\mathcal{C}^*(w))$ for the set of positions in $\mathcal{C}$ whose vertices are in $\mathcal{C}^*(w)$ excluding w.

For any regular set of positions, W, let

$$K(\mathcal{C}^*(W)) = \bigcup_{w \in W} K(\mathcal{C}^*(w))$$

Definition 30 *Call a set of positions, W,* simple *if*

1. *W is $\mathcal{C}$-regular*
2. *there exists a partition of the set $K(\mathcal{C}^*(W))$ into $K^\alpha(\mathcal{C}^*(W))$ and $K^\beta(\mathcal{C}^*(W))$ called* active *and* background *positions respectively such that*
   (a) *two background positions w_1 and w_2 are in the same stage if for all $w_1^1, \ldots, w_1^n \in K^\beta(\mathcal{C}^*(W))$ along a root-to-w_1 path there exist $w_2^1, \ldots, w_2^n \in K^\beta(\mathcal{C}^*(W))$ along a root-to-w_2 path such that the colour of the edge with parent w_1^i equals that of w_2^i for $i = 1, \ldots, n$*
   (b) *the same holds for the active positions and moreover if for w_1^n, w_2^n, $w_1, w_2 \in K^\alpha(\mathcal{C}^*(W))$, in the notation above, there exist the edges (w_1^n, w_1) and (w_2^n, w_2), then they have the same colour.*

Point 2. in Definition 30 means that two background positions are in the same stage if they share the same sequence of background edges. Note that it is sometimes tedious but always straightforward to determine from the coloured graph of $\mathcal{C}$ whether or not a given set of positions W is simple.

Definition 31 *A manipulation is called* amenable forcing to a set W *if*

1. *the set W is simple in $(\mathcal{C}, \Pi(\mathcal{C}))$,*
2. *the set W is simple in $(\mathcal{C}, \widehat{\Pi}(\mathcal{C}))$ and under $(\mathcal{C}, \widehat{\Pi}(\mathcal{C}))$, $\widehat{P}(W) = 1$,*
3. *$\Pi(\mathcal{C})$ and $\widehat{\Pi}(\mathcal{C})$ differ only on edges whose parents lie in $K^\alpha(\mathcal{C}^*(W))$.*

When $W = \{w\}$, a singleton, the set of background positions will be empty and so all the conditions above are vacuous and so W is simple. It follows that a pure manipulation forced to w is amenable. The point of Definitions 30 and 31 is that, in a sense to be defined below, the random variables associated with positions lying in $K^\alpha(C^*(W))$, are independent of those lying in $K^\beta(C^*(W))$. An amenable manipulation may change probabilities in active positions, but will always leave probabilities associated with variables labelled by background positions unchanged.

Thus remember $\pi(\Lambda^-(w_0, w))$ and $\widehat{\pi}(\Lambda^-(w_0, w))$ represent respectively the probabilities in the idle $(\mathcal{C}, \Pi(\mathcal{C}))$ and manipulated $(\mathcal{C}, \widehat{\Pi}(\mathcal{C}))$ that a path in C will pass through the position $w \in W$, that is it reaches w. From Equation 1 and Definition 30, for each $w \in W$

$$\pi(\Lambda^-(w_0, w)) = \pi^\alpha(\Lambda^-(w_0, w))\pi^\beta(\Lambda^-(w_0, w))$$

where $\pi^\alpha(\Lambda^-(w_0, w))$ $[\pi^\beta(\Lambda^-(w_0, w))]$ is a product of primitive probabilities in $\Pi(\mathcal{C})$ associated with random variables whose positions lie in $K^\alpha(C^*(W))$

$[K^\beta(C^*(W))]$ respectively. Furthermore, from the definition of $K^\alpha(C^*(W))$ for any indices $w, w' \in W$

$$\pi^\alpha(\Lambda^-(w_0, w)) = \pi^\alpha(\Lambda^-(w_0, w')) = \pi^\alpha_W \text{ (say)}$$

The fact that W is also simple in $(\mathcal{C}, \widehat{\Pi}(\mathcal{C}))$ for the amenable manipulation implies that

$$\widehat{\pi}(\Lambda^-(w_0, w)) = \widehat{\pi}^\alpha_W \ \widehat{\pi}^\beta(\Lambda^-(w_0, w))$$

for all $w \in W$. So summarising these comments, for an amenable manipulation we have that

$$\pi(\Lambda^-(w_0, w)) = \pi^\alpha_W \ \pi^\beta(\Lambda^-(w_0, w)) \tag{4}$$

and

$$\widehat{\pi}(\Lambda^-(w_0, w)) = \widehat{\pi}^\alpha_W \ \pi^\beta(\Lambda^-(w_0, w)) \tag{5}$$

Lemma 32 *Consider an amenable manipulation forcing to a simple set W. The distribution of $\widehat{Y}(W)$ – defined above in (2) – is identified from the probabilities in the unmanipulated system of the events $\{Y(W) = y, W\}$ for $y \in \Omega_Y$ and its probabilities are given by the equation*

$$\widehat{P}(\widehat{Y}(W) = y) = \frac{P(Y(W) = y, W)}{P(W)}$$

where $P(W) = \sum_{w \in W} \pi(\Lambda^-(w_0, w))$ and provided that $\pi(\Lambda^-(w_0, w)) > 0$ for all $w \in W$.

Proof.

$$\begin{aligned}
\widehat{P}(\widehat{Y} = y, W) &= \widehat{\pi}(\Lambda_y(w_0, W)) \\
&= \sum_{w \in W} \widehat{\pi}(\Lambda^+_y(w, w_\infty))\widehat{\pi}(\Lambda^-(w_0, w)) \\
&= \sum_{w \in W} \widehat{P}(\widehat{Y}(w) = y)\widehat{\pi}(\Lambda^-(w_0, w)) \qquad \text{by the definition of } \widehat{Y}(w) \\
&= \sum_{w \in W} P(Y(w) = y|w)\widehat{\pi}(\Lambda^-(w_0, w)) \qquad \text{by Lemma 28} \\
&= \sum_{w \in W} P(Y(w) = y|w)\widehat{\pi}^\alpha_W \pi^\beta(\Lambda^-(w_0, w))
\end{aligned}$$

by equation 5. Hence by Equation (4)

$$\begin{aligned}
\widehat{P}(Y(W) = y, W) &= \frac{\widehat{\pi}^\alpha_W}{\pi^\alpha_W} \sum_{w \in W} P(Y(w) = y|w)\pi^\alpha_W \pi^\beta(\Lambda^w) \\
&= \frac{\widehat{\pi}^\alpha_W}{\pi^\alpha_W}\pi(\Lambda(y, W)) = \frac{\widehat{\pi}^\alpha_W}{\pi^\alpha_W} P(Y(W) = y, W)
\end{aligned}$$

Since as a function of y

$$\widehat{P}(Y(W) = y, W) \propto P(Y(W) = y, W)$$

it follows that

$$\widehat{P}(\widehat{Y}(W) = y) = \frac{P(Y(W) = y, W)}{P(W)}$$

as required. ■

In a causal BN the effect of a manipulation of X on a later ordered random variable Y is identified from observing the distribution of the unmanipulated pair (X, Y) if and only if the vector of unobserved (hidden) variables $\mathbf{H}$ in the system can be partitioned as $\mathbf{H} = (\mathbf{H}_1, \mathbf{H}_2)$ where

$$\mathbf{H}_2 \perp\!\!\!\perp (\mathbf{H}_1, X)$$

and

$$(Y, \mathbf{H}_2) \perp\!\!\!\perp \mathbf{H}_1 | X$$

It is straightforward to check that, for a CEG drawn taking positions in any order associated with such a BN, this is exactly the condition of Lemma 32. In this correspondence the states of the vector of hidden variables $\mathbf{H}_1$ and X define the values the active positions take whilst the vector of hidden variables $\mathbf{H}_2$ define the values the background positions take. So Lemma 32 is an exact analogue of this well known result for causal BNs for the more general class of CEGs. Moreover conditions in Lemma 32 only depend on an appropriate factorisation of probabilities associated with the manipulated set W.

5.2. A Backdoor Theorem for CEG's

An important graphical condition on causal BNs, called the backdoor criterion, gives sufficient conditions for when values of a vector Z of measurements together with a manipulated variable X and an effect variable Y are observed but all other variables in the BN are hidden [27, Section 3.3.1] and [28]. We finish this section by generalising this result. We find an analogous theorem that applies a graphical and sufficient criterion to a CEG to determine whether we can identify the effect of an observed manipulation on a random variable Y from the observation of a random variable Z – happening before the manipulation in the partial ordering induced by the paths – together with the observation of Y in the unmanipulated system.

Our strategy is to apply the graphical results of Section 5.1. Let Z be a random variable observed in the unmanipulated network whose events $\{Z = z\}$, for $z \in \Omega_Z$ can be expressed as a partition $\{\Omega_z : z \in \Omega_Z\}$ of the set of positions. Suppose that there exists a fine cut Ω which gives a refinement of such partition. Let the set of paths in the unmanipulated CEG intersecting Ω_z be denoted by Λ_z, $z \in \Omega_Z$. For Z to occur before the manipulation we require that every position w, whose associated random variable is manipulated, lies on a path in the unmanipulated CEG between a position in Ω_z and w_∞, $z \in \Omega_Z$.

Note that under this condition, the probability of $\{Z = z\}$, $z \in \Omega_Z$, is the same under the manipulated CEG and the unmanipulated CEG. For $z \in \Omega_Z$

let $\mathcal{C}(\Omega_z)$ be the CEG defined in the last section. Furthermore the effect random variable Y defined in the last section is such that $\{Y = y|Z = z\}$ is a measurable event with respect to $\mathcal{C}(\Omega_z)$, $z \in \Omega_Z$ and

$$P(Y = y) = \sum_{z \in \Omega_Z} P(Y = y|Z = z)P(Z = z)$$

Definition 33 *A set of positions W in a CEG is called* simple conditioned on Z, *if*

1. $W = \bigcup_{z \in \Omega_Z} W(z)$ *where $W(z)$ is simple in $\mathcal{C}(\Omega_z)$,*
2. *each set $W(z)$ is non empty and contains, say, w_z, and*
3. *there is a directed path in $\mathcal{C}$ – and hence by definition also in $\mathcal{C}(\Omega_z)$ – from a position in Ω_z to w_z.*

Note that a simple W is simple conditioned on the constant function. It is possible to determine from the coloured graph of $\mathcal{C}$ whether or not W is simple conditioned on Z.

For $z \in \Omega_Z$ let $\mathcal{C}(\Omega_z)$ denote the CEG constructed in the same way as $\mathcal{C}(W)$ defined Section 5.1.2 and call the new root variable $X(w_0^*(z))$.

Consider an amenable manipulation to a set W and let W be simple conditioned on Z. Z is called a *backdoor variable* to the manipulation. Note that such manipulation does not change any primitive probabilities from the idle system lying on a path between w_0 and positions in Ω_z, $z \in \Omega_Z$. Let $\widehat{Y}(W)$ be the image of Y in the manipulated CEG. Then $\{\widehat{Y}(W) = y|Z = z\}$ is a $\mathcal{C}(\Omega_z)$ measurable random variable and

$$\widehat{P}(\widehat{Y}(W) = y) = \sum_{z \in \Omega_Z} \widehat{P}(\widehat{Y}(W) = y|Z = z)\widehat{P}(Z = z)$$

Theorem 34 *If a set W is simple conditioned on Z then the distribution of Y after an amenable manipulation to W and for which Z is a backdoor variable is identified from the probability (in the unmanipulated system) of the events $\{Y = y, W, Z = z\}$, $y \in \Omega_Y$, $z \in \Omega_Z$ and its probabilities are given by the formula*

$$\widehat{P}(\widehat{Y}(W) = y) = \sum_{z \in \Omega_Z} \frac{P(Y = y, W|Z = z)}{P(W|Z = z)} P(Z = z)$$

Proof. By definition

$$\widehat{P}(\widehat{Y}(W) = y, W|Z = z) = \widehat{P}(\widehat{Y}(W) = y, W(z)|Z = z)$$

where W is simple in $\mathcal{C}(\Omega_z)$, the CEG is valid given that $\{Z = z\}$. Applying exactly the same argument in the proof of Lemma 32 we have that

$$\widehat{P}(\widehat{Y}(W) = y, W(z)|Z = z) = \frac{P(Y(W) = y, W(z)|Z = z)}{P(W|Z = z)}$$

The result now follows. ■

Corollary 35 *Consider a causal CEG and a pure manipulation to a manipulation set W. Then if all the events $\{Y = y, W, Z = z\}$, $y \in \Omega_Y$, $z \in \Omega_Z$ are manifest, then the effect of the manipulation is identified and given by the formula above whenever W is simple conditioned on Z.*

Proof. It follows from Theorem 34 because the CEG is causal, so that the manipulation above is valid. ■

Note that if a CEG of a BN is constructed so that the backdoor variables are introduced as early as possible compatibly with the ordering of the BN, then the conditions of Theorem 34 are satisfied for atomic intervention on a causal BN.

So to summarise: by examining the topology and colouring of the CEG it is possible to determine sufficient conditions for whether an effect of a causal manipulation can be identified from a given partial set of observations of the paths that units take through a network of simulators. We feel this result is a very significant generalisation of the Backdoor theorem for two reasons. First it applies to highly asymmetric models just as well as ones exhibiting the strong types of symmetry that can be coded by a BN. For example different values of Z could subsequently lead to quite different topologies in the CEG evoking different ways of satisfying the criteria of Theorem 34 for different configurations z. Of course if it is possible to fully express a model using a BN then this property is quite useless because of each configuration z leads to identical topologies of $\mathcal{C}(\Omega_z)$.

Second the search for an appropriate random variable Z, whose observation ensures identifiability, is not just restricted to subvectors of the original (non-descendant) measurement vectors. We can search over all *functions* of such measurements. For example in an asymmetric model we might chose the indicator on whether one of the non-descendant measurements took a particular value. Searching over functions of measurements to find the cheapest way of identifying the quantity of interest will often be of much greater value than simply searching over subsets of measurements. This will be particularly useful if those measurements have not yet been collected, or their parametrisations have been chosen by convention rather than because they reflect in some natural way the mechanism by which things happen.

It can be shown, by adjusting the methodology of Section 3, that the graphical deconstruction outlined in the proofs of the results above can also be used to guide the estimation of a total causal effect of a manipulation. However such an analysis is nearly always non-conjugate and beyond the scope of this paper. For related issues see [34].

6. Discussion

CEGs provide a much more flexible and general framework within which to express ideas about causal relationships in discrete simulator networks than a BN. This is not an accident. For networks of simulators causality is more naturally expressed through predictions concerning the manipulation of unfolding

situations than it is through assertions about the effects of manipulations on dependence relationships between measurements. But as for the BN, a CEG is appropriate for expressing causal relationships in practical scenarios only when it might be plausible to believe that an observer's perceived reality can be fully expressed by this simulator analogy.

How often can a modeller reasonably believe that reality corresponds to manipulating some of the input settings of some of the simulators? Clearly this depends on the context. One might expect the analogy to work best when simulators are of physical components of a machine. Even in these contexts the simulator analogy can be fragile. For example, it is common practice when designing a car to search for an optimal design by simulating various components, performing computer experiments and pasting the results together in ways similar to those described above. But only after a prototype car is actually built, the inadequacy of the models based on this network of simulators can become apparent.

Of course if randomised experiments are designed so that what is observed exactly mirrors a system later to be manipulated, then the simulator analogy is almost automatically sound when used to answer what might happen to a typical unit [22, 23]. Essentially such experiments attempt to isolate a component of the real system modelled by a single simulator in the network and address policy questions about the relevant population under specific types of unambiguous manipulation. But whenever the simulators are networked together to produce a composite picture and are used as a source of data, enormous leaps of faith have to be made. We believe that to label such speculative deductions as "causal" deductions introduces an implicit spurious determinism which could be inappropriate.

We have limited the scope of the models discussed in this paper intentionally. Now we turn to briefly discuss two generalisations.

First it is commonplace to meet structures that cannot be expressed simply in terms of the exchangeable relationships in a BN, as in the examples above. Many examples of when no BN can fully describe a structure are given in, for example, [30, 31]. We have found in these cases that the partial description of the model through the CEG is helpful for framing causal hypotheses. However issues of estimation and identifiability are subsequently better addressed through combining graphical methods with the algebraic structure of the model. Various methods for exploiting the combined algebraic and graphical structures of a CEG model to address estimation and identification will be reported on a later paper.

The prior densities we have considered in this paper have purposely been chosen as naive: in any practical context it can be expected that non-modular context specific information will have to be incorporated and a numerical Bayesian methodology will typically be needed. Under the conditions in Section 3 above the general methodology we have described above is still valid.

Despite these caveats we hope we have demonstrated the advantages of the CEG over the BN as a framework for expressing processes where predictions

about classes of manipulations need to be made and explained. They are much more general than the BN, are often more simple to explain and, under the appropriate assumptions as easy to estimate. Most importantly we would argue that, just because data is conveyed to us within a certain parametrisation this should not be allowed to force us to think of potential causal hypotheses only in terms of these random variables as encouraged by the BN technology. It is now well appreciated that it is often necessary to separate causal structure from the dependence structures introduced into measurements through a particular sampling mechanism specific to the acquisition of information for a particular study. The BN is not the most transparent framework within which to accomplish this separation. Indeed in our experience it is not an expressive enough framework within which to accomplish this task. The CEG however does provide such a framework.

References

[1] P. Anderson and J.Q. Smith (2005). A graphical framework for representing the semantics of asymmetric models. *Technical report* 05-12, CRiSM, *Department of Statistics*, The University of Warwick.

[2] T. Bedford and R. Cooke (2001). *Probabilistic risk analysis: foundations and methods*. Cambridge University Press, Cambridge.

[3] C. Boutilier, N. Freidman, M. Goldszmidt and D. Koller (1996). Context-specific independence in Bayesian networks. In *Proceedings of UAI - 96* 115-123.

[4] R.E. Bryant (1986). Graphical algorithms for Boolean function manipulation. *IEEE Transactions of Computers* C-35 677-691.

[5] R.J. Castelo (2002). *The Discrete Acyclic Digraph Markov Model in Data Mining*. Utrecht University, Ph.D. thesis.

[6] R. Castelo and M.D. Perlman (2004). Learning essential graph Markov models from data. In A. Gamze, S. Moral and A. Salmeron (eds.) *Advances in Bayesian networks*, Springer, Berlin, 255–269.

[7] G.A. Churchill (1995). Accurate restoration of DNA sequences. In C. Gatsaris et al. (eds.) *Case Studies in Bayesian Statistics vol. II*, Springer-Verlag, New-York, 90-148.

[8] D. Cooper and C. Yoo (1999). Causal discovery from a mixture of experimental and observational data. In K.B. Laskey and H. Prade (eds.) *Proceedings of the Seventeen Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, San Francisco.

[9] R.G. Cowell, A.P. Dawid, S.L. Lauritzen and D.J. Spiegelhalter (1999). *Probabilistic Networks and Expert Systems*, Springer-Verlag, New York.

[10] A.P. Dawid (2000). Causality without counterfactuals *J. Amer. Statist. Ass.* 95:407-448.

[11] D.G.T. Denison, C.C. Holmes, B.K. Mallick and A.F.M. Smith (2002). *Bayesian methods for nonlinear classification and regression*, John Wiley & Sons Ltd., Chichester.

[12] N. Freidman and M. Goldszmidt (1999). Learning Bayesian networks

with local structure. In M.I. Jordan (ed.) *Learning in Graphical Models* MIT Press, 421-459.

[13] S. French (ed.) (1989). *Readings in Decision analysis.* Chapman and Hall/CRC, London.

[14] D. Glymour and G.F. Cooper (1999). *Computation, Causation, and Discovery.* MIT Press, Cambridge, MA.

[15] D. Hausman (1998). *Causal Asymmetries*, Cambridge University Press, Cambridge.

[16] P.W. Holland (1986). Statistics and causal inference (With discussion and a reply by the author). *Journal of the American Statistical Association*, 81(396):945–970.

[17] M.I. Jordan (ed.) (1999). *Learning in Graphical Models*, MIT Press, Cambridge, MA.

[18] M. Jaeger (2004). Probabilistic decision graphs - combining verification and AI techniques for probabilistic inference. *Int.J. of Uncertainty, Fuzziness and Knowledge-based Systems*, 12:19-42.

[19] S.L. Lauritzen (1996). *Graphical models.* Oxford Science Press, Oxford, 1st edition.

[20] R. Lyons (1990). Random walks and percolation on trees. *Annals of Probability*, 18:931-958.

[21] D. Madigan and A. E. Raftery (1994). Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association* 89(428): 1535-1546.

[22] A.M. Madrigal and J.Q. Smith (2004). Causal Identification in Design Networks. In Sucar et al. (eds.) *Advances in Artificial Intelligence 2*, Springer-Verlag, 517-526.

[23] A.M. Madrigal (2004). *Evaluations of Policy Interventions under Experimental Conditions using Bayesian Influence Diagrams.* The University of Warwick, Ph.D. Thesis.

[24] D. McAllester, M. Collins and F. Pereira (2004). Case-factor diagrams for structured probability models. In *Proceedings of UAI - 2004* 382-391.

[25] J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems.* Morgan Kauffman, San Mateo.

[26] J. Pearl (1995). Causal diagrams for empirical research. *Biometrika,* 82:669-710.

[27] J. Pearl (2000). *Causality. models, reasoning and inference.* Cambridge University Press, Cambridge.

[28] J. Pearl (2003). Statistics and Causal Inference: A Review (with discussion). *Test*, 12(2):281-345.

[29] D. Poole and N.L. Zhang (2003). Exploiting contextual independence in probabilistic inference. *Journal of Artificial Intelligence research*, 18:263-313.

[30] E. Riccomagno and J.Q. Smith (2003). Non-Graphical Causality: a generalisation of the concept of a total cause. *Research report series No. 394, Dept of Statistics,* The University of Warwick.

[31] E. Riccomagno and J.Q. Smith (2004). Identifying a cause in models which are not simple Bayesian networks. In *Proceedings of the 10th Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 1315 -1322.

[32] J.M. Robins (1986). A new approach to causal inference in mortality studies with a sustained exposure period —application to control of the healthy worker survivor effect. *Mathematical Modeling*, 7:1393-1512.

[33] J.M. Robins (1997). Causal inference from complex longitudinal data. In M. Berkane (ed.) *Latent variable modeling and applications to causality* (Los Angeles, CA, 1994). Springer-Verlag, New York, 69–117.

[34] J.M. Robins, R. Scheines, P. Spirtes, and L. Wasserman (2003). Uniform Consistency in Causal Inference. *Biometrika* 90(3):491-515.

[35] D. Rubin (1973). Estimating causal effects of treatments in randomised and non - randomised studies. *J. Educational Psychology* 66:688-701.

[36] G. Shafer (1996). *The Art of Causal Conjecture*. MIT Press, Cambridge, MA.

[37] R. Settimi and J.Q. Smith (1998). On the geometry of Bayesian graphical models with hidden variables. In G. Cooper and S. Moral (eds.) *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers, S. Francisco, 472–479.

[38] R. Settimi and J.Q. Smith (2000). Geometry, moments and conditional independence trees with hidden variables. *The Annals of Statistics*, 28(4):1179-1205.

[39] J.Q. Smith, A.E. Faria, S. French, D. Ranyard, J. Bohunova, T. Duranova, M. Stubna, L. Dutton, C. Rojas and A. Sohier (1997). Probabilistic data assimilation within RODOS. *Radiation Protection Dosimetry* 73(1-4):57-59.

[40] J.Q. Smith and E. E. Anderson (2006). Conditional independence and Chain Event Graphs. *Artificial Intelligence*, to appear.

[41] J.Q. Smith and J. Croft (2003). Bayesian networks for discrete multivariate data: An algebraic approach to inference. *J. of Multivariate Analysis*, 84(2):387-402.

[42] D. Spiegelhalter, A.P. Dawid, S.L. Lauritzen and R.G. Cowell (1993). Bayesian analysis of expert systems. *Statistical Science*, 8:219-282.

[43] P. Spirtes, C. Glymour and R. Scheines (1993). *Causation, Prediction, and Search.* Springer-Verlag, New York.

Corso Duca degli Abruzzi 24
Turin 10129, Italy
printeade1

Gibbet Hill Road
Coventry CV4 7AL, UK
printeade1